Introductory Programming Using Python

Day 2

Seow Khee Wei

Source code: http://bit.ly/2vXKZIL

Introduction of trainer

Name

Seow Khee Wei

Telegram

@kwseow

Email

seow_khee_wei@rp.edu.sg

Programme Day Two

Morning

- Read and writing files
- Copying, moving and deleting files and folders
- Working with Excel
- Processing CSV files

Afternoon

- Image processing
- Connecting to the Web
- Sending emails

File Paths

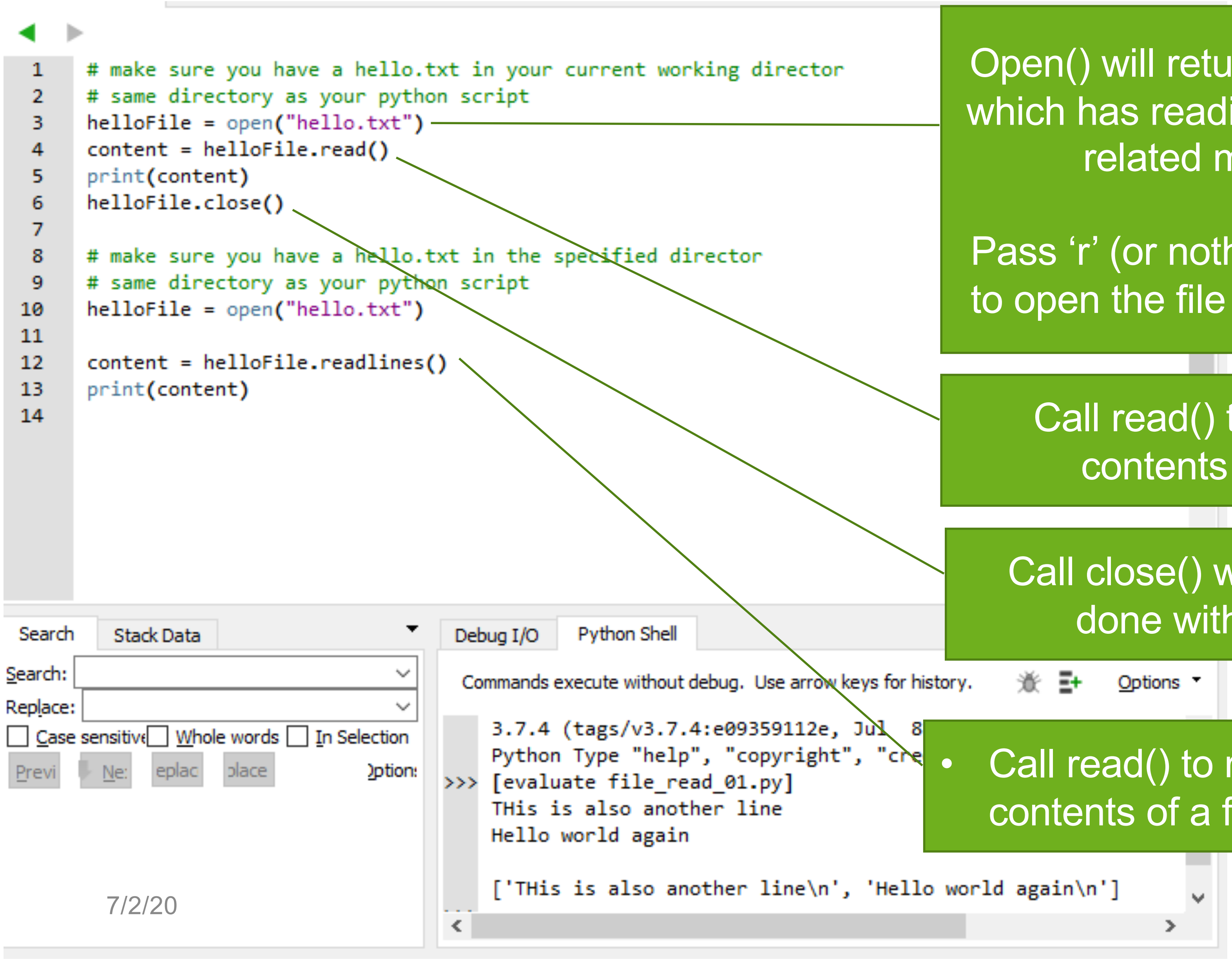
Absolute file paths are notated by a **leading forward slash or drive label**.
For example, `/home/example_user/example_directory` or `C:/system32/cmd.exe`

An absolute file path describes how to access a given file or directory, starting from the root of the file system. A file path is also called a *pathname*.

Relative file paths are notated by a **lack of a leading forward slash**.
For example, `example_directory`.

A relative file path is interpreted from the perspective your current working directory. If you use a relative file path from the wrong directory, then the path will refer to a different file than you intend, or it will refer to no file at all..

Read files



```
# make sure you have a hello.txt in your current working director
# same directory as your python script
helloFile = open("hello.txt")
content = helloFile.read()
print(content)
helloFile.close()

# make sure you have a hello.txt in the specified director
# same directory as your python script
helloFile = open("hello.txt")

content = helloFile.readlines()
print(content)

```

```
3.7.4 (tags/v3.7.4:e09359112e, Jul  8
Python Type "help", "copyright", "cre
>>> [evaluate file_read_01.py]
THis is also another line
Hello world again

['THis is also another line\n', 'Hello world again\n']
```

Open() will return a file object which has reading and writing related methods

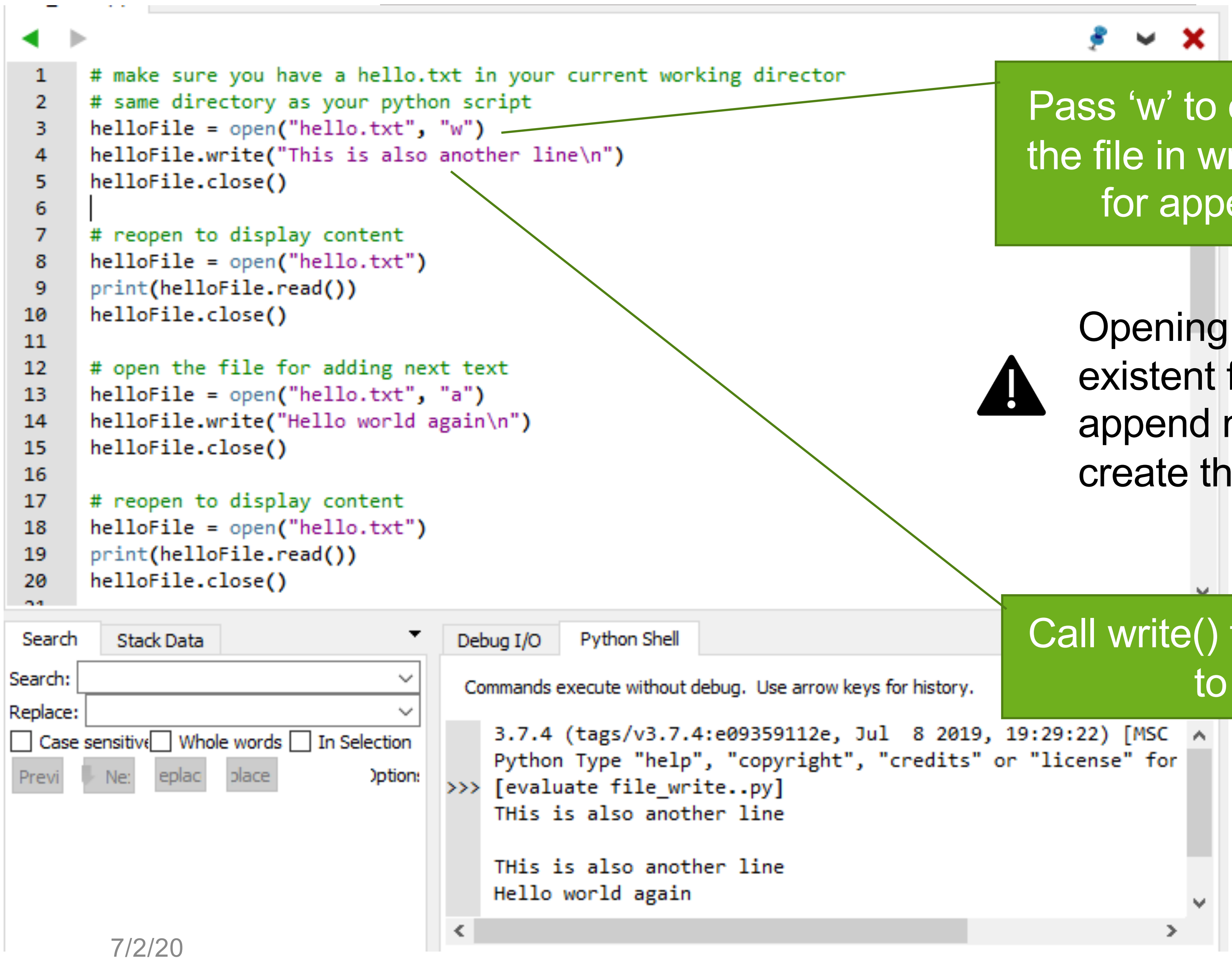
Pass ‘r’ (or nothing) to open() to open the file in read mode.

Call read() to read the contents of a file

Call close() when you are done with the file.

- Call read() to read the contents of a file

Write files



```
# make sure you have a hello.txt in your current working director
# same directory as your python script
helloFile = open("hello.txt", "w")
helloFile.write("This is also another line\n")
helloFile.close()

# reopen to display content
helloFile = open("hello.txt")
print(helloFile.read())
helloFile.close()

# open the file for adding next text
helloFile = open("hello.txt", "a")
helloFile.write("Hello world again\n")
helloFile.close()

# reopen to display content
helloFile = open("hello.txt")
print(helloFile.read())
helloFile.close()
```

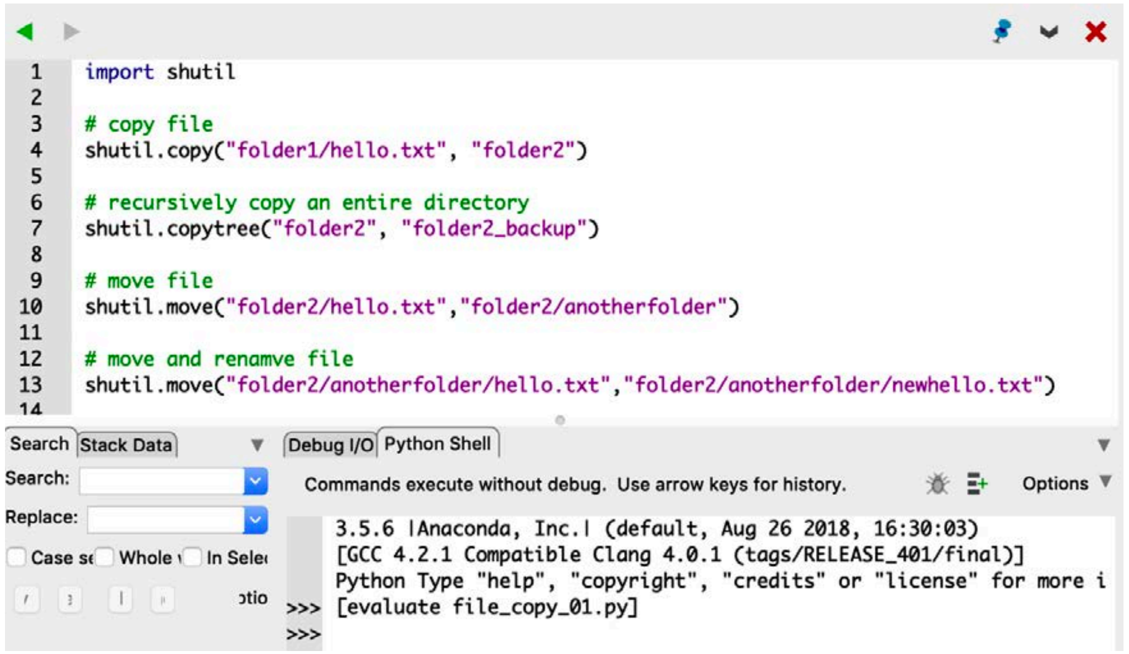
```
Commands execute without debug.  Use arrow keys for history.

    3.7.4 (tags/v3.7.4:e09359112e, Jul  8 2019, 19:29:22) [MSC
    Python Type "help", "copyright", "credits" or "license" for
>>> [evaluate file_write..py]
    THis is also another line

    THis is also another line
    Hello world again
```

Pass ‘w’ to open() to open the file in write mode or ‘a’ for append mode.

⚠ Opening a non-existent file in write or append mode will create that file

Call write() to write a string to a file.

Copy and moving files



```
import shutil

# copy file
shutil.copy("folder1/hello.txt", "folder2")

# recursively copy an entire directory
shutil.copytree("folder2", "folder2_backup")

# move file
shutil.move("folder2/hello.txt","folder2/anotherfolder")

# move and renamve file
shutil.move("folder2/anotherfolder/hello.txt","folder2/anotherfolder/newhello.txt")
```

Commands execute without debug. Use arrow keys for history.

```
    3.5.6 |Anaconda, Inc.| (default, Aug 26 2018, 16:30:03)
    [GCC 4.2.1 Compatible Clang 4.0.1 (tags/RELEASE_401/final)]
    Python Type "help", "copyright", "credits" or "license" for more i
>>> [evaluate file_copy_01.py]
>>>
```

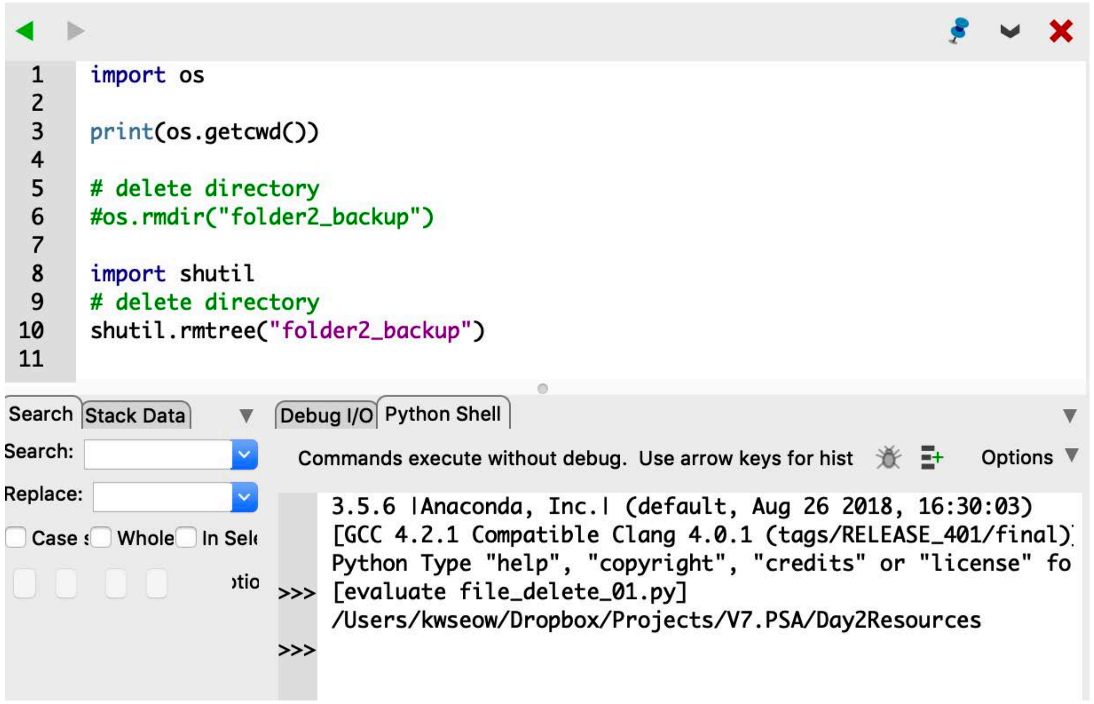
- shutil.copy(src, dst) – Copy the file *src* to the file or directory *dst*

- shutil.copytree(src, dst) - Recursively copy an entire directory tree rooted at *src*.

- shutil.move(src, dst) - Recursively move a file or directory (*src*) to another location (*dst*).

https://docs.python.org/3/library/shutil.html

Deleting files

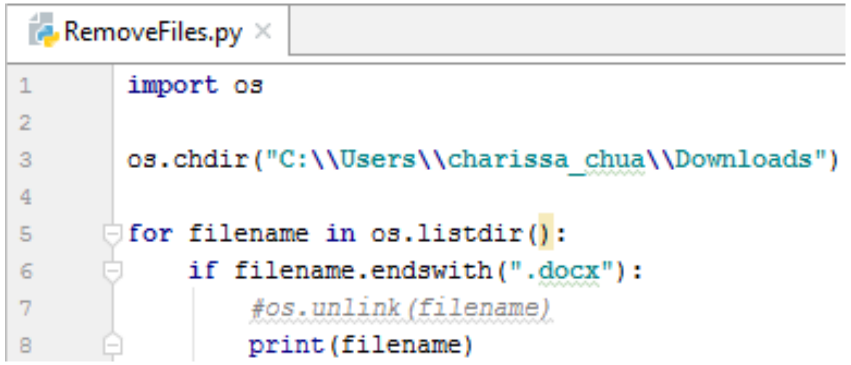


```
import os

print(os.getcwd())

# delete directory
#os.rmdir("folder2_backup")

import shutil
# delete directory
shutil.rmtree("folder2_backup")

```

Commands execute without debug. Use arrow keys for hist

```
    3.5.6 |Anaconda, Inc.| (default, Aug 26 2018, 16:30:03)
    [GCC 4.2.1 Compatible Clang 4.0.1 (tags/RELEASE_401/final)
    Python Type "help", "copyright", "credits" or "license" fo
>>> [evaluate file_delete_01.py]
    /Users/kwseow/Dropbox/Projects/V7.PSA/Day2Resources

>>>
```

- os.unlink() will delete a file
- os.rmdir() will delete a folder (but folder must be empty)
- shutil.rmtree() will delete a folder and all its contents



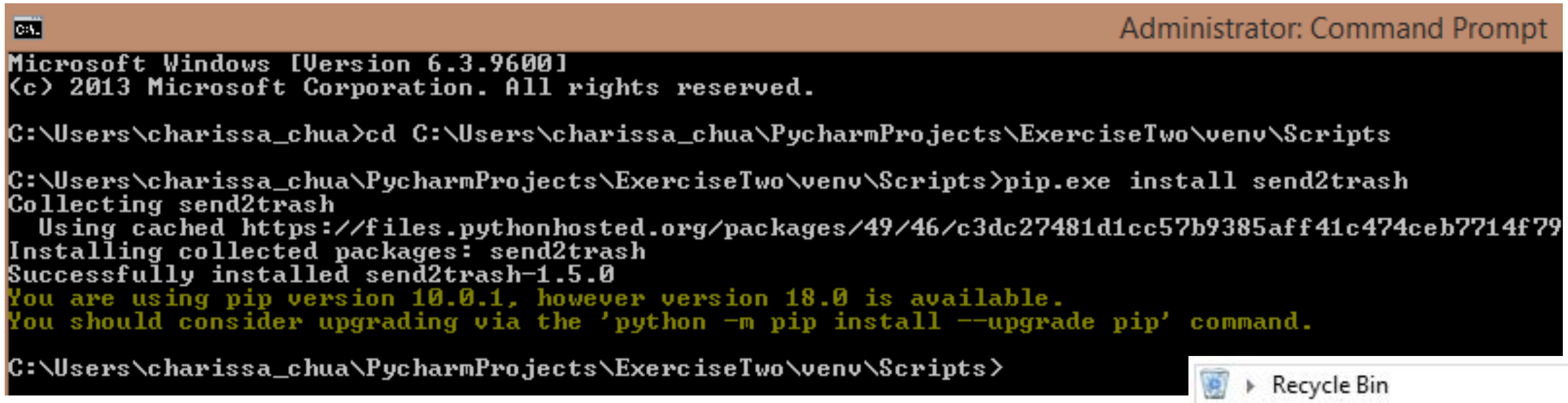
RemoveFiles.py

```
import os

os.chdir("C:\\Users\\charissa_chua\\Downloads")

for filename in os.listdir():
    if filename.endswith(".docx"):
        #os.unlink(filename)
        print(filename)
```

⚠ Deleting can be dangerous, so do a dry run first

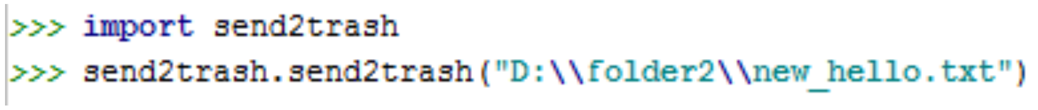
send2Trash module

- Install send2trash module using pip.exe
- send2trash.send2trash() will send a file or folder to the recycling bin

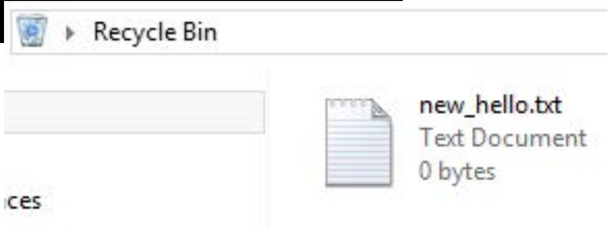


Administrator: Command Prompt

```
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\charissa_chua>cd C:\Users\charissa_chua\PycharmProjects\ExerciseTwo\venv\Scripts

C:\Users\charissa_chua\PycharmProjects\ExerciseTwo\venv\Scripts>pip.exe install send2trash
Collecting send2trash
  Using cached https://files.pythonhosted.org/packages/49/46/c3dc27481d1cc57b9385aff41c474ceb7714f79
Installing collected packages: send2trash
Successfully installed send2trash-1.5.0
You are using pip version 10.0.1, however version 18.0 is available.
You should consider upgrading via the 'python -m pip install --upgrade pip' command.

C:\Users\charissa_chua\PycharmProjects\ExerciseTwo\venv\Scripts>
```



```
>>> import send2trash
>>> send2trash.send2trash("D:\\folder2\\new_hello.txt")
```



Walk a directory to perform some tasks

```
D:\animals
|   animals.txt
|
+---cats
|       cute_kitten.jpg
|
\---dogs
    |   dogs.txt
    |
    \---retriever
            golden-retriever.jpg
```

```python
import os

for folderName, subfolders, filenames in os.walk('D:\\animals'):
    print('The current folder is ' + folderName)

    for subfolder in subfolders:
        print('SUBFOLDER OF ' + folderName + ': ' + subfolder)
    for filename in filenames:
        print('FILE INSIDE ' + folderName + ': '+ filename)

    print('')
```

```
Python Type "help", "copyright", "credits" or "license" for m
[evaluate dir_walk.py]
The current folder is D:\animals
SUBFOLDER OF D:\animals: cats
SUBFOLDER OF D:\animals: dogs
FILE INSIDE D:\animals: animals.txt

The current folder is D:\animals\cats
FILE INSIDE D:\animals\cats: cute_kitten.jpg

The current folder is D:\animals\dogs
SUBFOLDER OF D:\animals\dogs: retriever
FILE INSIDE D:\animals\dogs: dogs.txt

The current folder is D:\animals\dogs\retriever
FILE INSIDE D:\animals\dogs\retriever: golden-retriever.jpg
```

os.walk()

The os.walk() function is passed a single string value: the path of a folder. You can use os.walk() in a for loop statement to walk a directory tree, much like how you can use the range() function to walk over a range of numbers. Unlike range(), the os.walk() function will return three values on each iteration through the loop:

- A string of the current folder’s name
- A list of strings of the folders in the current folder
- A list of strings of the files in the current folder

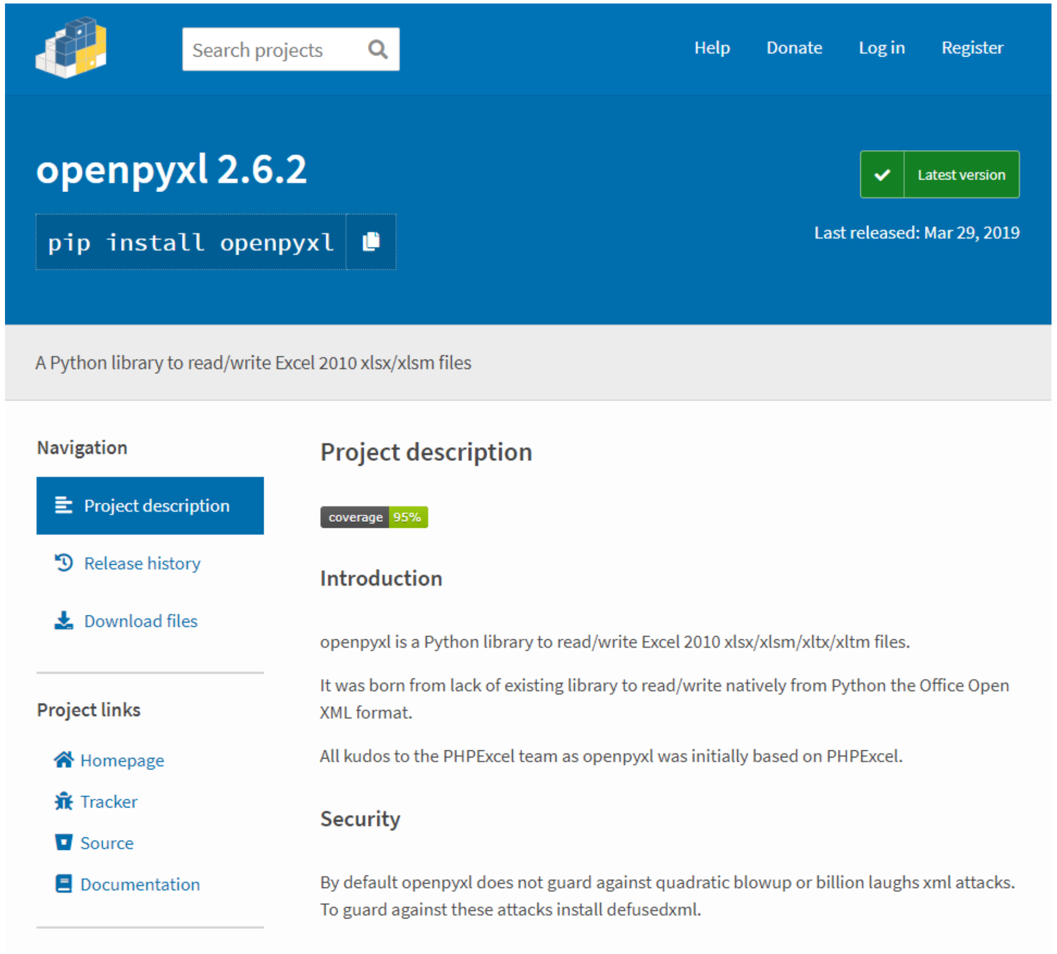
(By current folder, we mean the folder for the current iteration of the for loop. The current working directory of the program is not changed by os.walk().)

Exercise 1

- Write a script to list all the files in the C:\Users directory

Working with Excel

- Install openpyxl module using "pip install openpyxl"
- Make sure the file is available - students_attendance.xlsx
- Full openpyxl documentation: https://openpyxl.readthedocs.io/en/stable/index.html



openpyxl 2.6.2

```
pip install openpyxl
```

Last released: Mar 29, 2019

A Python library to read/write Excel 2010 xlsx/xlsm files

Project description

Introduction

openpyxl is a Python library to read/write Excel 2010 xlsx/xlsm/xltx/xltm files.

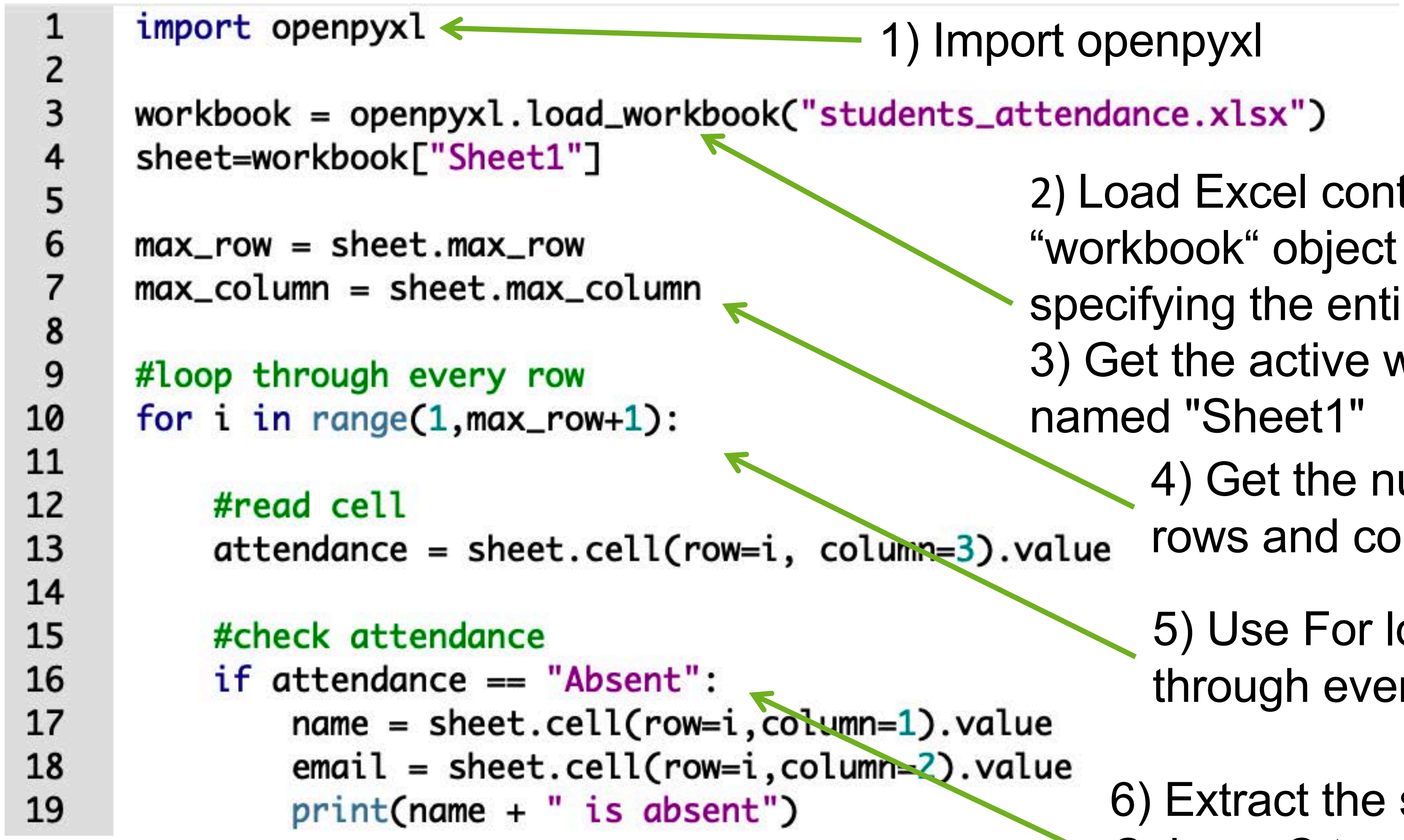
It was born from lack of existing library to read/write natively from Python the Office Open XML format.

All kudos to the PHPExcel team as openpyxl was initially based on PHPExcel.

Security

By default openpyxl does not guard against quadratic blowup or billion laughs xml attacks. To guard against these attacks install defusedxml.

Reading Excel file



```
import openpyxl

workbook = openpyxl.load_workbook("students_attendance.xlsx")
sheet=workbook["Sheet1"]

max_row = sheet.max_row
max_column = sheet.max_column

#loop through every row
for i in range(1,max_row+1):

    #read cell
    attendance = sheet.cell(row=i, column=3).value

    #check attendance
    if attendance == "Absent":
        name = sheet.cell(row=i,column=1).value
        email = sheet.cell(row=i,column=2).value
        print(name + " is absent")
```

1) Import openpyxl

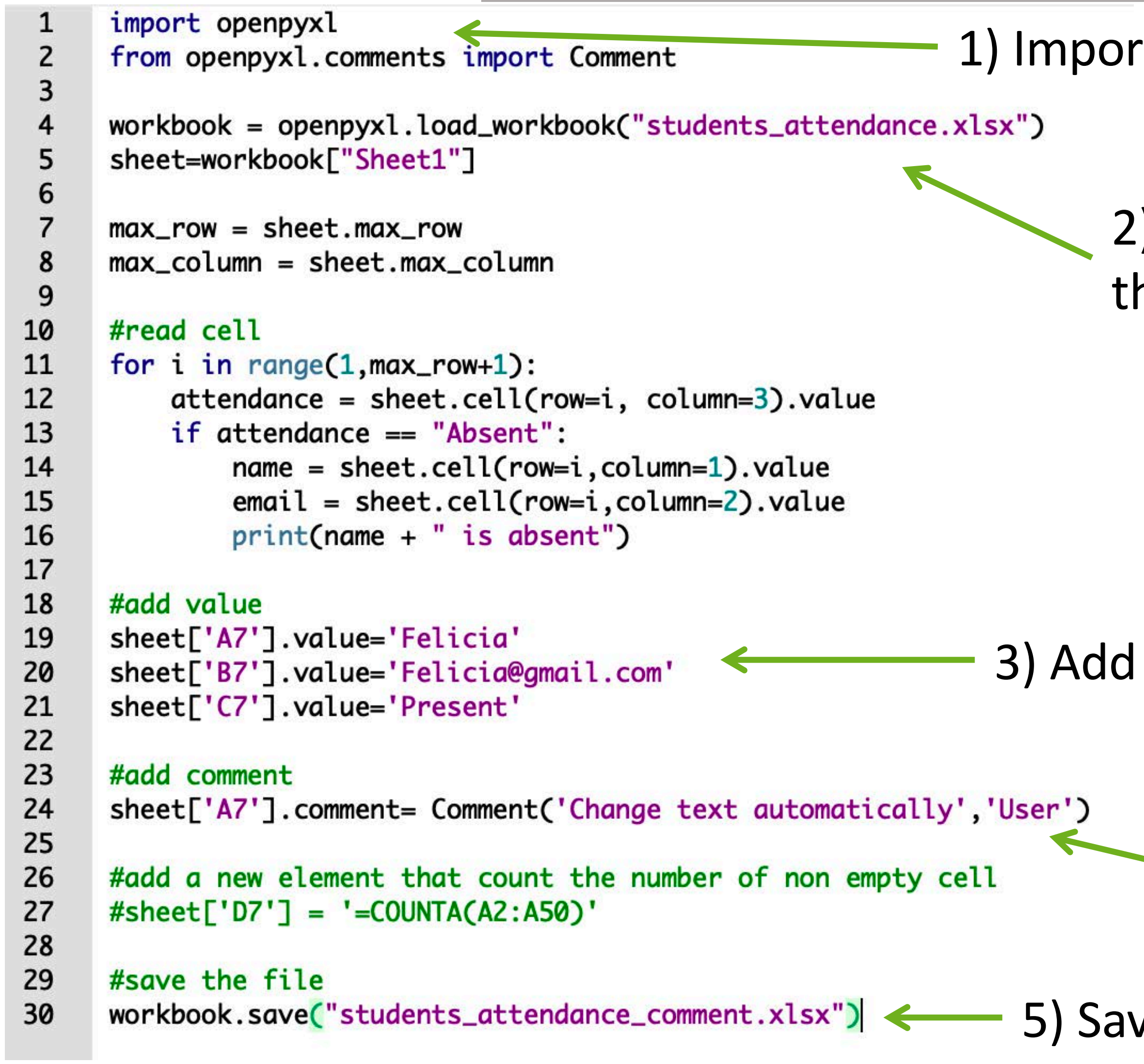
2) Load Excel content into “workbook“ object by specifying the entire path
3) Get the active worksheet named "Sheet1"

4) Get the number of rows and columns

5) Use For loop to go through every row

6) Extract the status at Column C to check for attendance

Update Excel file



```python
import openpyxl
from openpyxl.comments import Comment

workbook = openpyxl.load_workbook("students_attendance.xlsx")
sheet=workbook["Sheet1"]

max_row = sheet.max_row
max_column = sheet.max_column

#read cell
for i in range(1,max_row+1):
    attendance = sheet.cell(row=i, column=3).value
    if attendance == "Absent":
        name = sheet.cell(row=i,column=1).value
        email = sheet.cell(row=i,column=2).value
        print(name + " is absent")

#add value
sheet['A7'].value='Felicia'
sheet['B7'].value='Felicia@gmail.com'
sheet['C7'].value='Present'

#add comment
sheet['A7'].comment= Comment('Change text automatically','User')

#add a new element that count the number of non empty cell
#sheet['D7'] = '=COUNTA(A2:A50)'

#save the file
workbook.save("students_attendance_comment.xlsx")
```

1) Import openpyxl

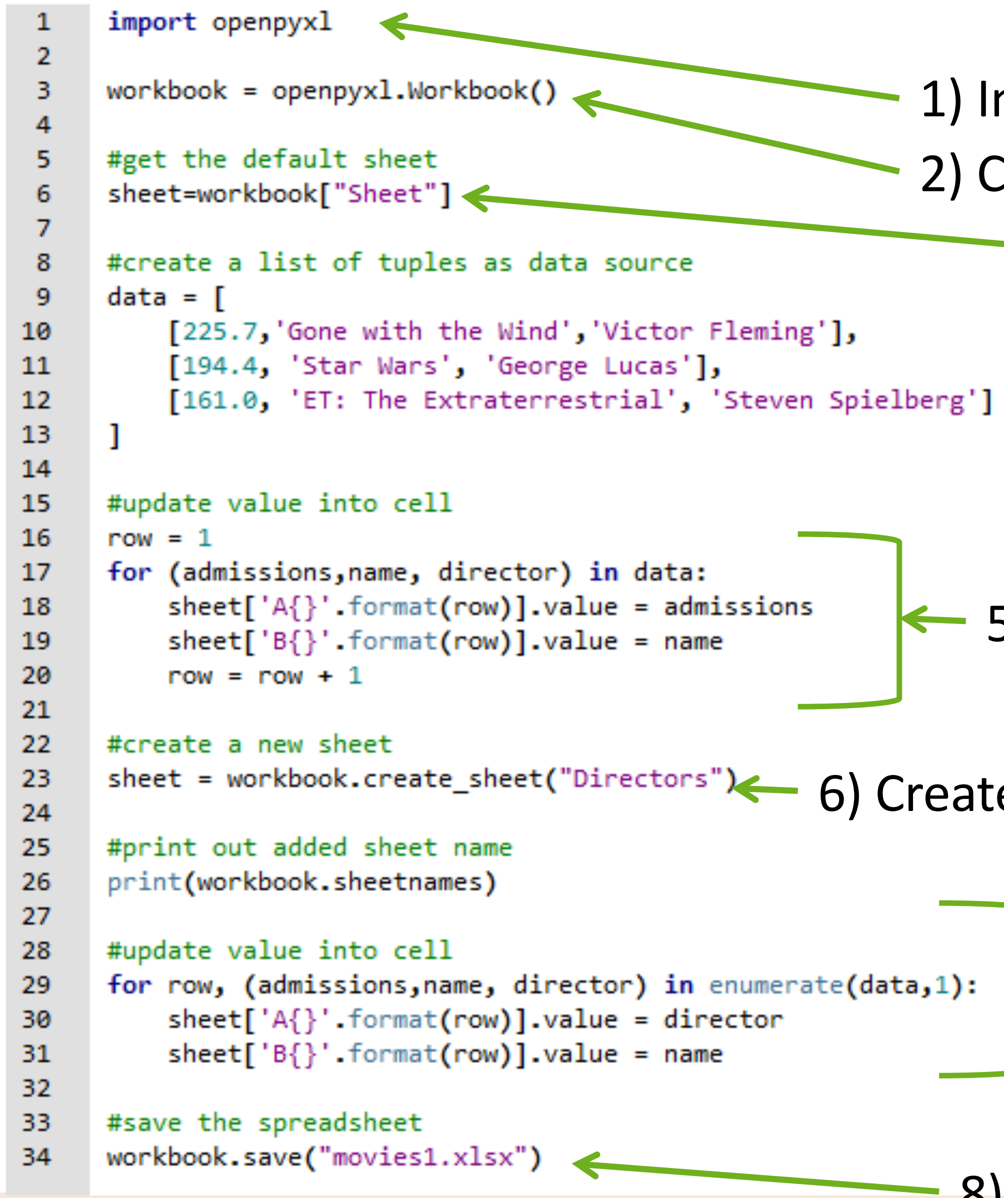
2) Load file into memory & get the sheet

3) Add value to cell

4) Add comments to cell

5) Save the spreadsheet

Create Excel file



```
import openpyxl

workbook = openpyxl.Workbook()

#get the default sheet
sheet=workbook["Sheet"]

#create a list of tuples as data source
data = [
    [225.7,'Gone with the Wind','Victor Fleming'],
    [194.4, 'Star Wars', 'George Lucas'],
    [161.0, 'ET: The Extraterrestrial', 'Steven Spielberg']
]

#update value into cell
row = 1
for (admissions,name, director) in data:
    sheet['A{}'.format(row)].value = admissions
    sheet['B{}'.format(row)].value = name
    row = row + 1

#create a new sheet
sheet = workbook.create_sheet("Directors")

#print out added sheet name
print(workbook.sheetnames)

#update value into cell
for row, (admissions,name, director) in enumerate(data,1):
    sheet['A{}'.format(row)].value = director
    sheet['B{}'.format(row)].value = name

#save the spreadsheet
workbook.save("movies1.xlsx")
```

1) Import openpyxl

2) Create new workbook

3) Get default sheet

4) Create dataset - a list of lists

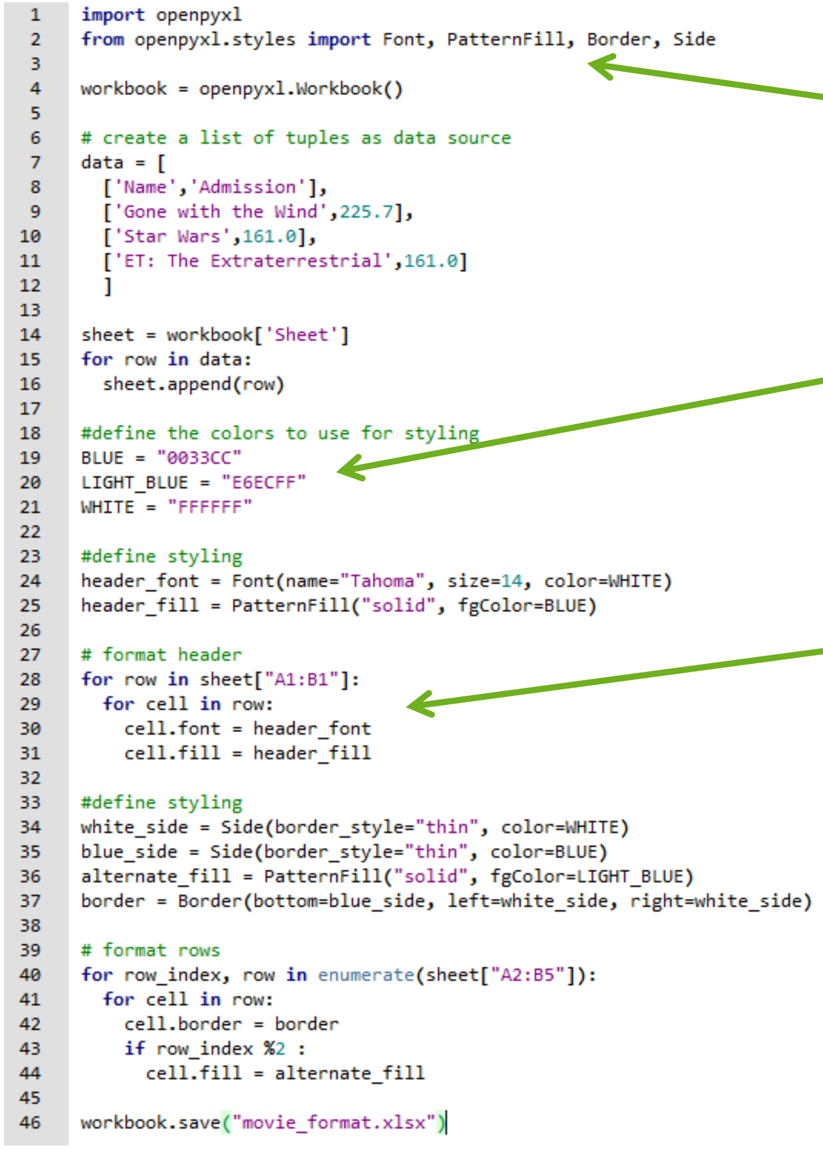
5) Insert value into cells

6) Create a new sheet

7) Insert value into cells

8) Save the spreadsheet

Format Excel



```python
import openpyxl
from openpyxl.styles import Font, PatternFill, Border, Side

workbook = openpyxl.Workbook()

# create a list of tuples as data source
data = [
  ['Name','Admission'],
  ['Gone with the Wind',225.7],
  ['Star Wars',161.0],
  ['ET: The Extraterrestrial',161.0]
  ]

sheet = workbook['Sheet']
for row in data:
  sheet.append(row)

#define the colors to use for styling
BLUE = "0033CC"
LIGHT_BLUE = "E6ECFF"
WHITE = "FFFFFF"

#define styling
header_font = Font(name="Tahoma", size=14, color=WHITE)
header_fill = PatternFill("solid", fgColor=BLUE)

# format header
for row in sheet["A1:B1"]:
  for cell in row:
    cell.font = header_font
    cell.fill = header_fill

#define styling
white_side = Side(border_style="thin", color=WHITE)
blue_side = Side(border_style="thin", color=BLUE)
alternate_fill = PatternFill("solid", fgColor=LIGHT_BLUE)
border = Border(bottom=blue_side, left=white_side, right=white_side)

# format rows
for row_index, row in enumerate(sheet["A2:B5"]):
  for cell in row:
    cell.border = border
    if row_index %2 :
      cell.fill = alternate_fill

workbook.save("movie_format.xlsx")
```

Import necessary functions

Setup colors and styles

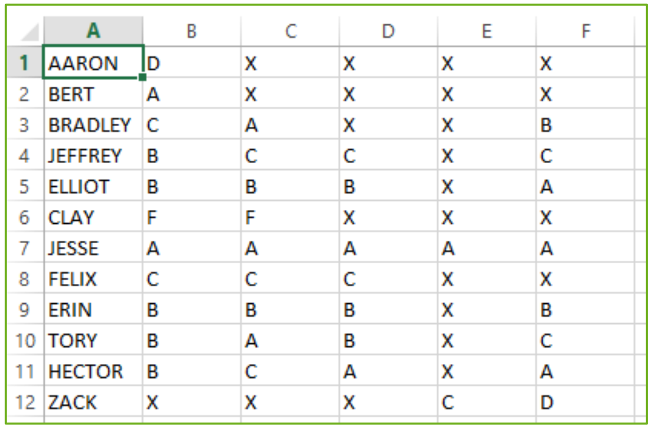
Loop through cell and set properties

Working with CSV file

- CSV stands for Comma-Separated Values (sometimes also called Comma Delimited File).
- It is commonly used for storing data in a table structured format.
- Each line/row in the file is a data record.
- Each field in the row is separated using a comma. The comma serves as a column boundary (aka delimiter) that separates the values into different cells of a table. (see next slide)

What is CSV format

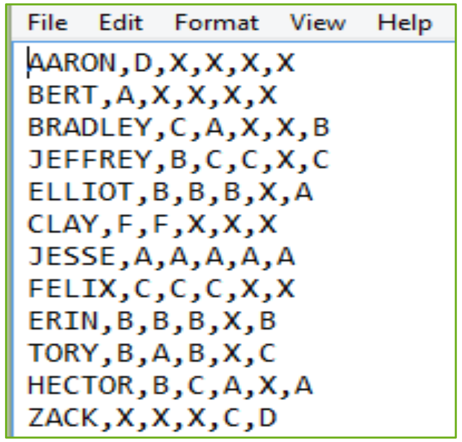
- The same data when viewed with Excel …



	A	B	C	D	E	F
1	AARON	D	X	X	X	X
2	BERT	A	X	X	X	X
3	BRADLEY	C	A	X	X	B
4	JEFFREY	B	C	C	X	C
5	ELLIOT	B	B	B	X	A
6	CLAY	F	F	X	X	X
7	JESSE	A	A	A	A	A
8	FELIX	C	C	C	X	X
9	ERIN	B	B	B	X	B
10	TORY	B	A	B	X	C
11	HECTOR	B	C	A	X	A
12	ZACK	X	X	X	C	D

Data is automatically tabulated in Excel into rows and columns (each value is in a cell)

- … and when viewed in plain text (e.g. in notepad) …



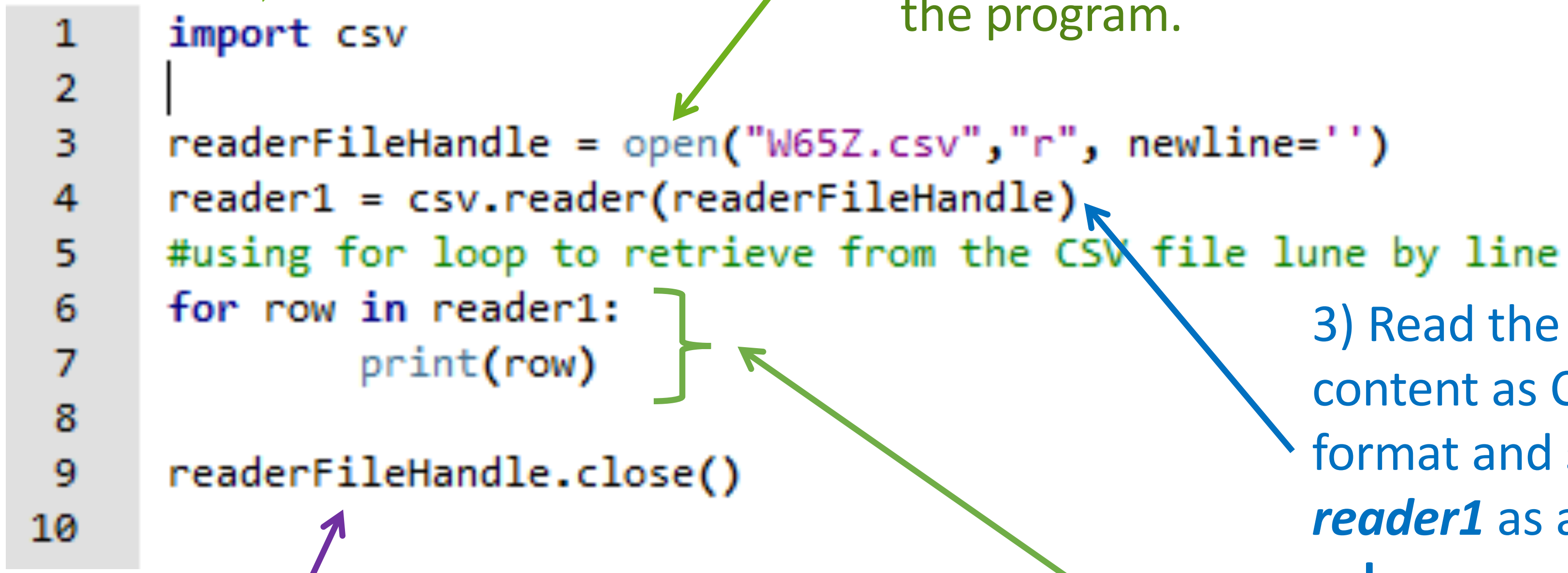
```
AARON,D,X,X,X,X
BERT,A,X,X,X,X
BRADLEY,C,A,X,X,B
JEFFREY,B,C,C,X,C
ELLIOT,B,B,B,X,A
CLAY,F,F,X,X,X
JESSE,A,A,A,A,A
FELIX,C,C,C,X,X
ERIN,B,B,B,X,B
TORY,B,A,B,X,C
HECTOR,B,C,A,X,A
ZACK,X,X,X,C,D
```

This is the RAW FORMAT of the file seen by computer programs:
- Each row is a record
- Values in a row are separated / delimited by comma ','

Reading CSV file

1) Load CSV library

2) Open the file named ‘W65Z.csv’ for reading (indicated by ‘r’ argument). ***readerFileHandle*** linked the CSV file to the program.



```
import csv

readerFileHandle = open("W65Z.csv","r", newline='')
reader1 = csv.reader(readerFileHandle)
#using for loop to retrieve from the CSV file lune by line
for row in reader1:
        print(row)

readerFileHandle.close()
```

3) Read the file content as CSV format and store it in ***reader1*** as a **list of values**

5) Close the file (remove the link)

4) For-loop retrieve each item in ***reader1*** **(a list)** into the loop variable ***row*** and display it. *(Note: Each line in the file becomes a list of values)*

Writing CSV file

1) Load CSV library

2) Create (if new) & Open the file named "**W65z_new.csv**" for writing (indicated by 'w' argument). ***writerFileHandle*** links the file to the program.

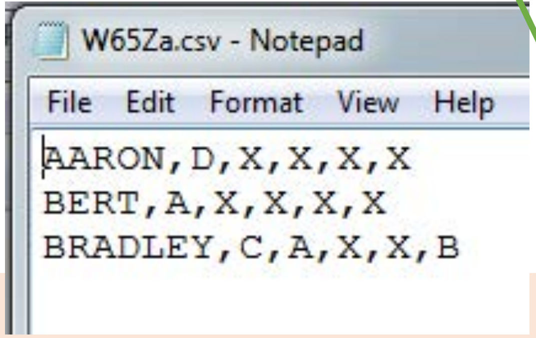
```python
import csv

writerFileHandle = open("new.csv", "w", newline='')
writer1 = csv.writer(writerFileHandle)
row1 = ["Arron","D","X","X","X","X"]
row2 = ["Bert","A","X","C","B","X"]
row3 = ["Bradley","C","A","C","X","X"]
rowlist = [row1,row2,row3]

for row in rowlist:
        writer1.writerow(row)

writerFileHandle.close()
```

3) "***writer1***" stores content to be written to the file in CSV format

4) ***rowlist*** stores the content to be written to the CSV file. (***rowlist*** is a list containing lists as items)

5) For-loop retrieves each item from ***rowlist*** into loop variable ***row*** → ***row*** (a list) is written as 1 csv formatted line into the file.

6) Close the file (Remove the link)



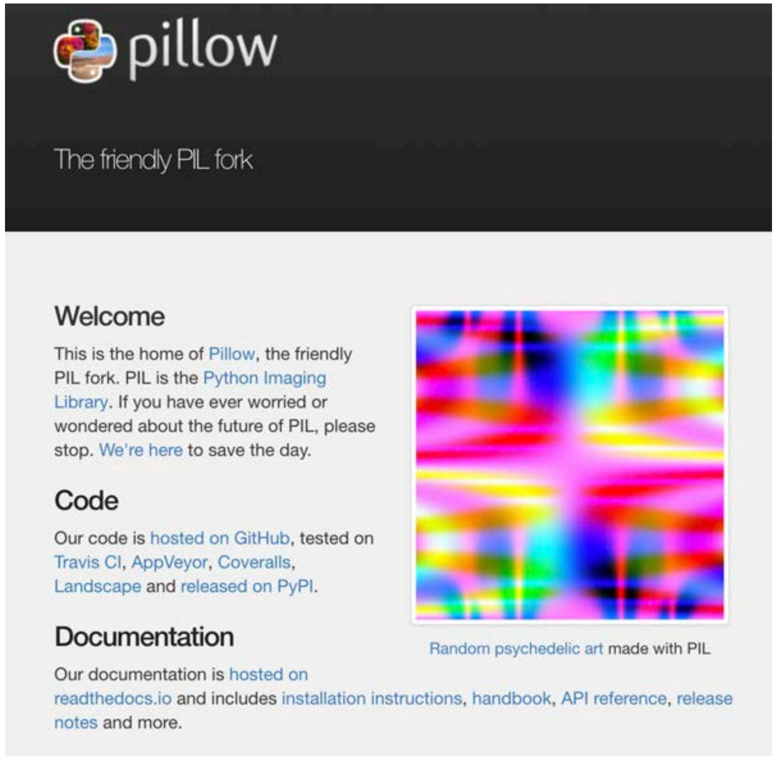
```
AARON,D,X,X,X,X
BERT,A,X,X,X,X
BRADLEY,C,A,X,X,B
```

Exercise 2

- Download the Annual Car Population by Make xls from https://www.mytransport.sg/content/mytransport/home/dataMall/static-data.html (or get a copy from trainer)
- Write a script that read this xls file and create a CSV file that contains only statistics for B.M.W and Honda

Quiz

Image Processing


pillow

The friendly PIL fork

Welcome

This is the home of Pillow, the friendly PIL fork. PIL is the Python Imaging Library. If you have ever worried or wondered about the future of PIL, please stop. We're here to save the day.

Code

Our code is hosted on GitHub, tested on Travis CI, AppVeyor, Coveralls, Landscape and released on PyPI.

Documentation

Our documentation is hosted on readthedocs.io and includes installation instructions, handbook, API reference, release notes and more.

Random psychedelic art made with PIL

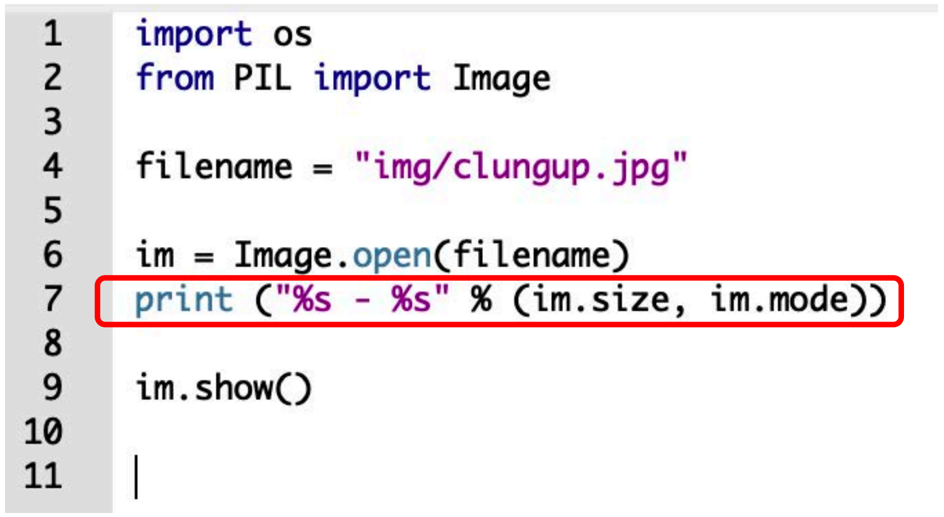
For the next section we are going to use the Python Image Library, or in short Pillow.

Install using the following command:
pip install Pillow

The documentation is at:
http://pillow.readthedocs.io/en/5.1.x/handbook/index.html

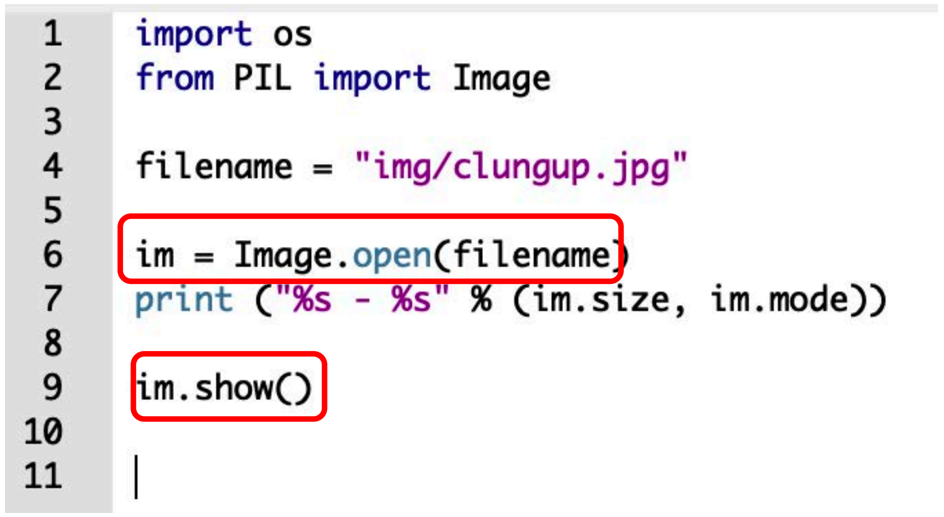
Image Processing

- Let’s print some info



```python
import os
from PIL import Image

filename = "img/clungup.jpg"

im = Image.open(filename)
print ("%s - %s" % (im.size, im.mode))

im.show()

```

Image Processing



```python
import os
from PIL import Image

filename = "img/clungup.jpg"

im = Image.open(filename)
print ("%s - %s" % (im.size, im.mode))

im.show()

```

Let’s explore what Pillow can do.

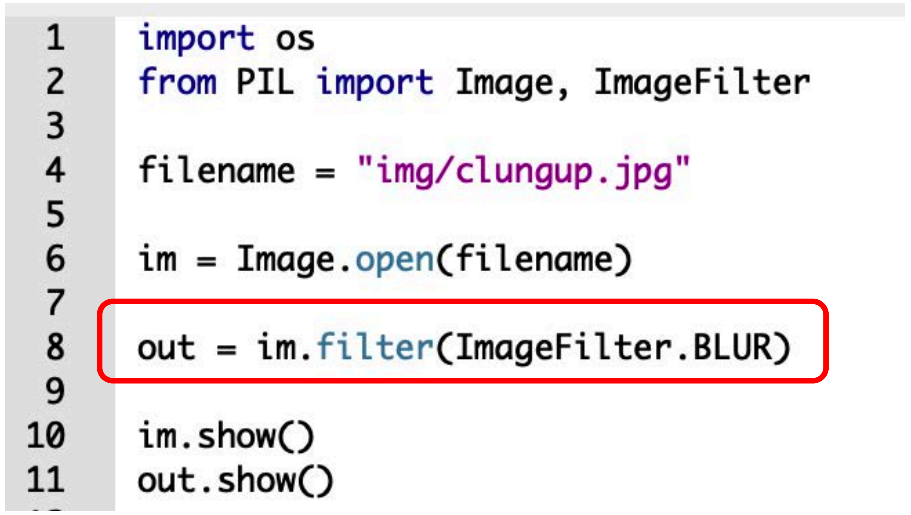
As a start we need to import it:

import Image

We can open images with
im = Image.open(fullname)

Then we can get the size of the image using im.size

Image Processing



```
import os
from PIL import Image, ImageFilter

filename = "img/clungup.jpg"

im = Image.open(filename)

out = im.filter(ImageFilter.BLUR)

im.show()
out.show()
```

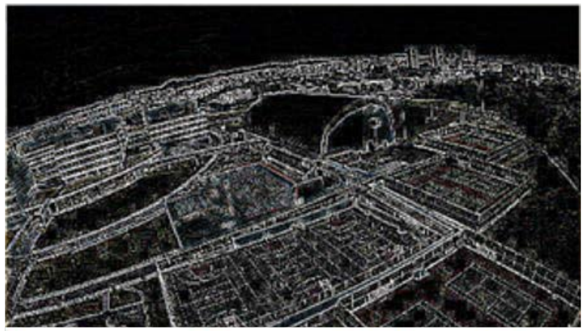
Now that we can load and understand the image, it is time to try and modify it.

Pillow has many conversion and filters, we will use some of them. But if you need more, go ahead : http://pillow.readthedocs.io/en/5.1.x/handbook/index.html

To use filters we need to extend our import:
from PIL import Image, ImageFilter

The way you can apply filters is :
out = im.filter(ImageFilter.BLUR)
Try some different filters!

Image processing - filters



```
image = image.filter(ImageFilter.FIND_EDGES)
```

```
image = ImageOps.grayscale(image)
```

```
image = ImageOps.solarize(image)
```

```
image = image.filter(ImageFilter.CONTOUR)
```

⚠ * Remember to include
ImageOps in your import statement

Image Processing - Rotating

Flipping the image horizontally or vertically

```
out = im.transpose(Image.FLIP_LEFT_RIGHT)
out = im.transpose(Image.FLIP_TOP_BOTTOM)
```

Rotating the image

```
out = im.transpose(Image.ROTATE_90)
out = im.transpose(Image.ROTATE_180)
out = im.transpose(Image.ROTATE_270)
```

Contrast

First add ImageEnhance to our imports:

```
from PIL import Image, ImageFilter, ImageEnhance
```

Then:

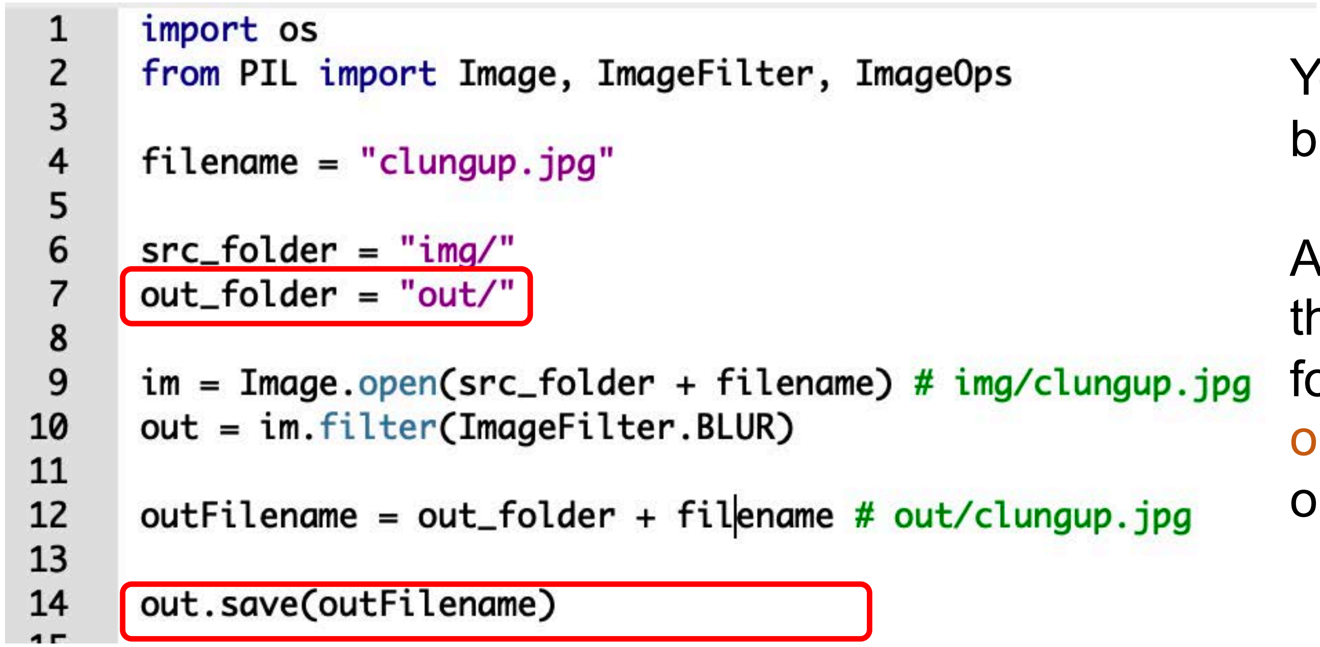
```
enh = ImageEnhance.Contrast(im)
out = enh.enhance(1.3)
```

We can do a lot with images.
Let's look at rotation and flipping

Try to rotate and flip your images.

Another cool effect is to make it brighter by changing the contrast

Image Processing - Writing

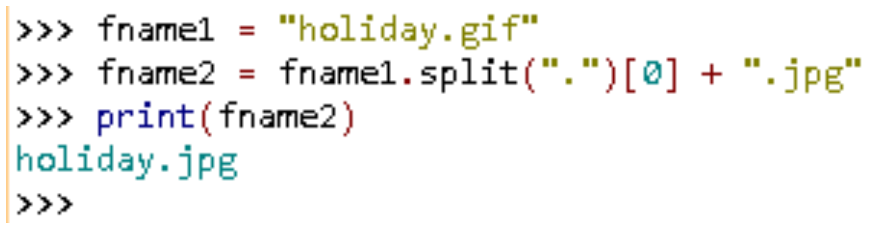


```
import os
from PIL import Image, ImageFilter, ImageOps

filename = "clungup.jpg"

src_folder = "img/"
out_folder = "out/"

im = Image.open(src_folder + filename) # img/clungup.jpg
out = im.filter(ImageFilter.BLUR)

outFilename = out_folder + filename # out/clungup.jpg

out.save(outFilename)
```

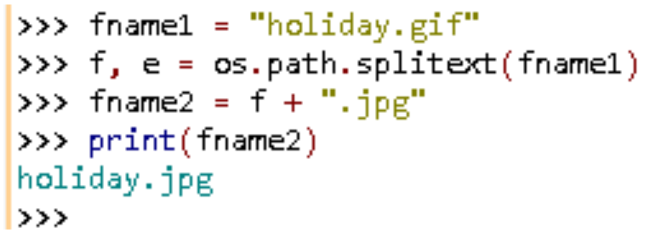
You can see the image, but it’s not being saved !

All you need to do to save the images in the “out" folder is: out.save(the name of the output file)

Image processing – Converting



```
>>> fname1 = "holiday.gif"
>>> fname2 = fname1.split(".")[0] + ".jpg"
>>> print(fname2)
holiday.jpg
>>>
```



```
>>> fname1 = "holiday.gif"
>>> f, e = os.path.splitext(fname1)
>>> fname2 = f + ".jpg"
>>> print(fname2)
holiday.jpg
>>>
```

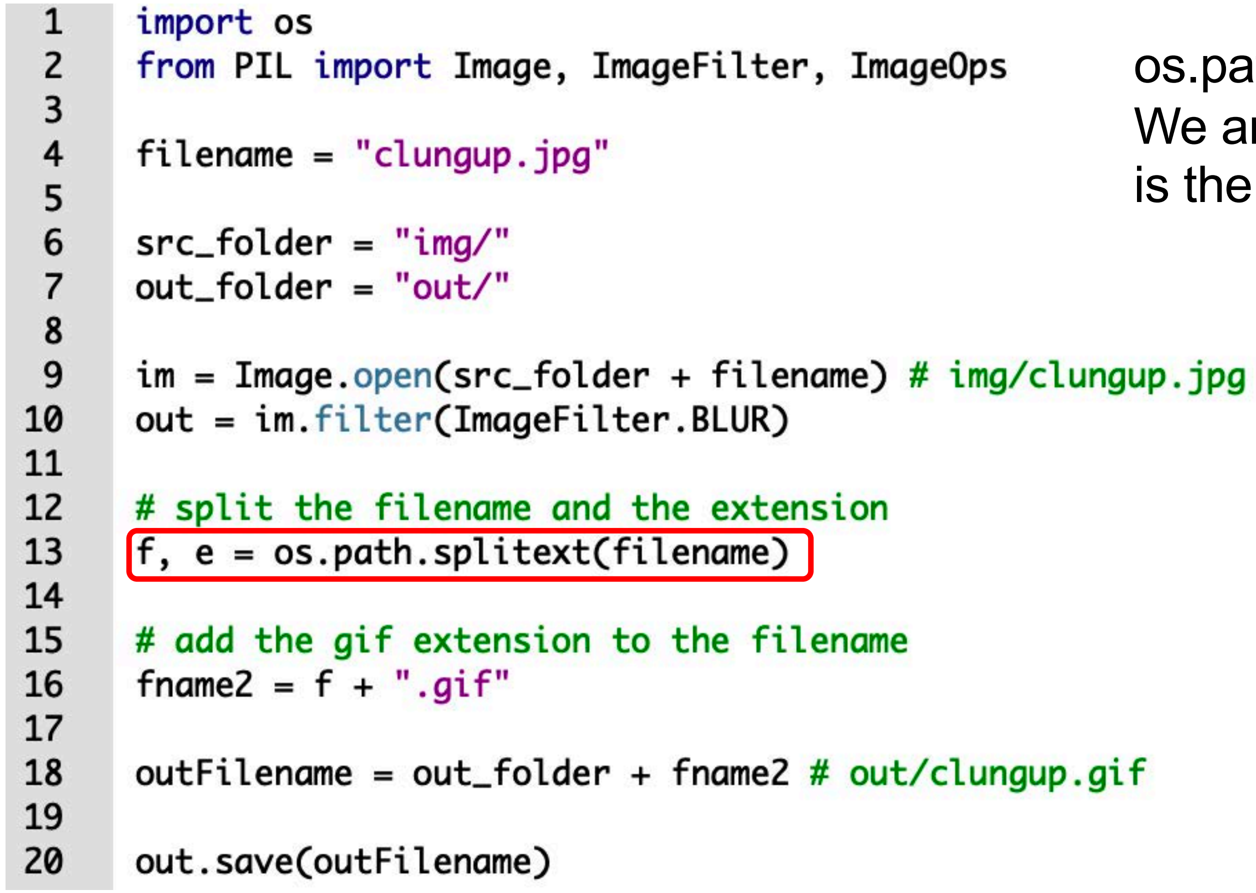
Maybe you want to keep all your photos in the same format.
We have some gif files and maybe you would have bmp or png images.

Pillow understands the output file, and will convert if the output file is different from the input.

fname1		fname2
holiday.gif	->	holiday.jpg

How can we convert the string holday.gif to holiday.jpg ?

Image processing – Converting

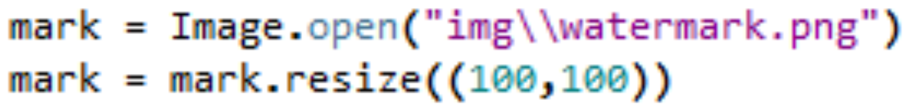


```python
import os
from PIL import Image, ImageFilter, ImageOps

filename = "clungup.jpg"

src_folder = "img/"
out_folder = "out/"

im = Image.open(src_folder + filename) # img/clungup.jpg
out = im.filter(ImageFilter.BLUR)

# split the filename and the extension
f, e = os.path.splitext(filename)

# add the gif extension to the filename
fname2 = f + ".gif"

outFilename = out_folder + fname2 # out/clungup.gif

out.save(outFilename)
```

os.path.splitext(file) returns a list. We are only interested in f, which is the first item in the list.

Image processing – Watermark

Create the mark image →
You can reduce the size to 100,100



```
mark = Image.open("img\\watermark.png")
mark = mark.resize((100,100))
```

Create a new function called

```
def watermark(im, mark, position):
        ….
```

It takes the original image, the watermark image and the desired position that we want the watermark to appear. The function will return the result.

We can use this function like:

```
watermark(im, mark, (0, 50)).show()
```

or

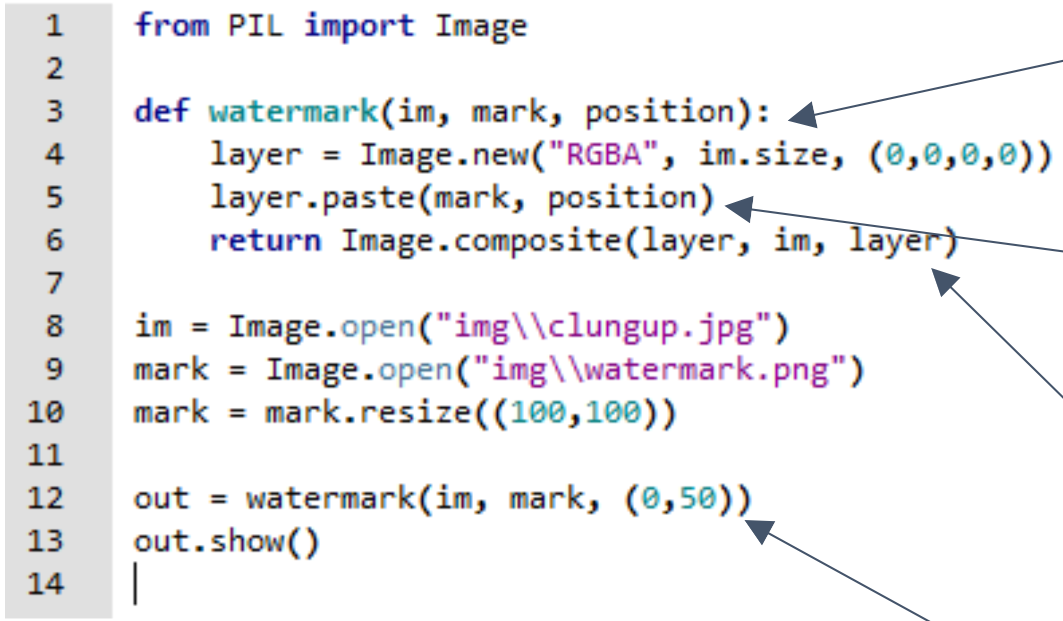
```
imOut = watermark(im, mark, (0,50))
imOut.save(fileOut)
```

Maybe you want to leave a small footprint on your images, called watermark.

In this case we can use the \\img\\watermark.png
and place it in each image on the bottom right.

Copyright
@RP

Image processing – Watermark



```python
from PIL import Image

def watermark(im, mark, position):
    layer = Image.new("RGBA", im.size, (0,0,0,0))
    layer.paste(mark, position)
    return Image.composite(layer, im, layer)

im = Image.open("img\\clungup.jpg")
mark = Image.open("img\\watermark.png")
mark = mark.resize((100,100))

out = watermark(im, mark, (0,50))
out.show()
```

First we need to create a new layer with the size of the original image.

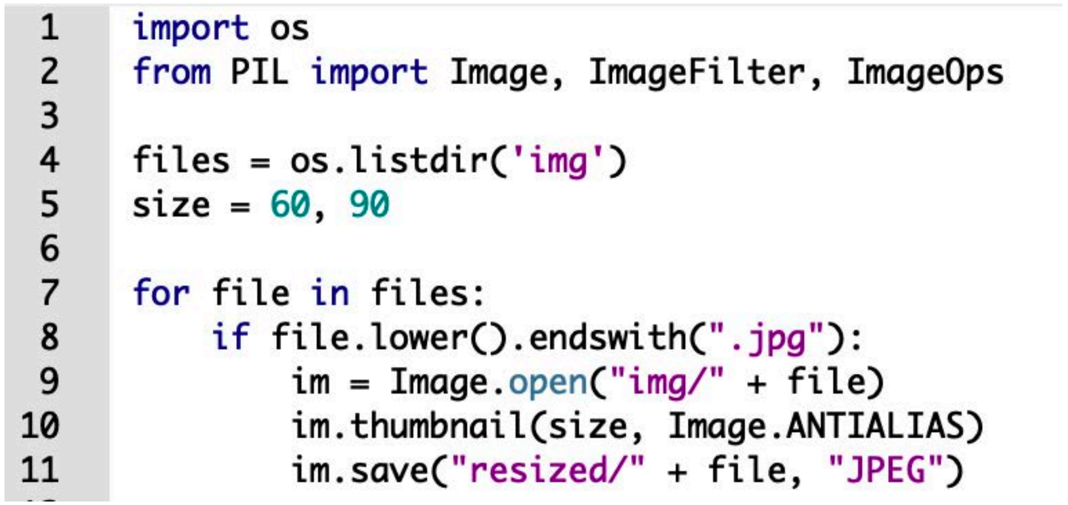
Then we paste the watermark image at the desired position and we return the composite.

Finally we merge the image and the layer together and return the result.

Then you can use it like this:

Batch Resize

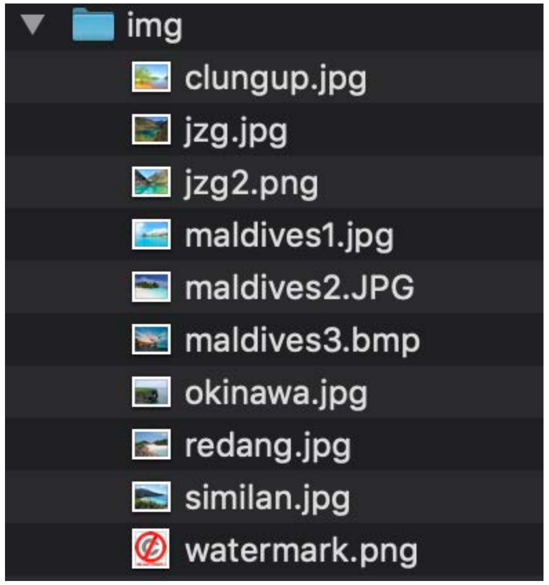
- Find all the files in “img” folder with “.jpg” extension
- Resize all the file to 60 x 90.
- Save all the files to the resized folder



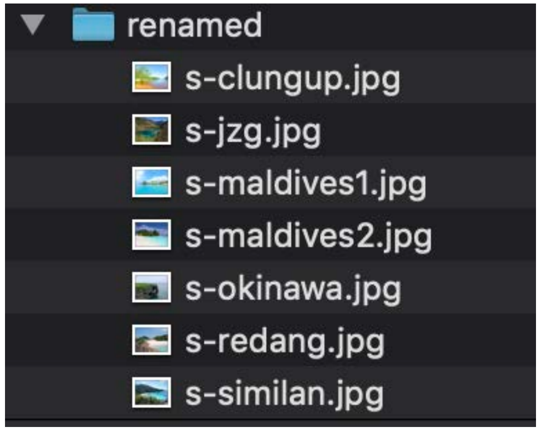
```python
import os
from PIL import Image, ImageFilter, ImageOps

files = os.listdir('img')
size = 60, 90

for file in files:
    if file.lower().endswith(".jpg"):
        im = Image.open("img/" + file)
        im.thumbnail(size, Image.ANTIALIAS)
        im.save("resized/" + file, "JPEG")
```

Exercise 3

- Batch Rename
  1. Find all the files in “img” folder with “.jpg” extension
  2. Copy all the files to a folder called ***renamed***
  3. Rename all the files with the “s-” prefix.



```
img
    clungup.jpg
    jzg.jpg
    jzg2.png
    maldives1.jpg
    maldives2.JPG
    maldives3.bmp
    okinawa.jpg
    redang.jpg
    similan.jpg
    watermark.png
```



```
renamed
    s-clungup.jpg
    s-jzg.jpg
    s-maldives1.jpg
    s-maldives2.jpg
    s-okinawa.jpg
    s-redang.jpg
    s-similan.jpg
```

Connecting to the Web

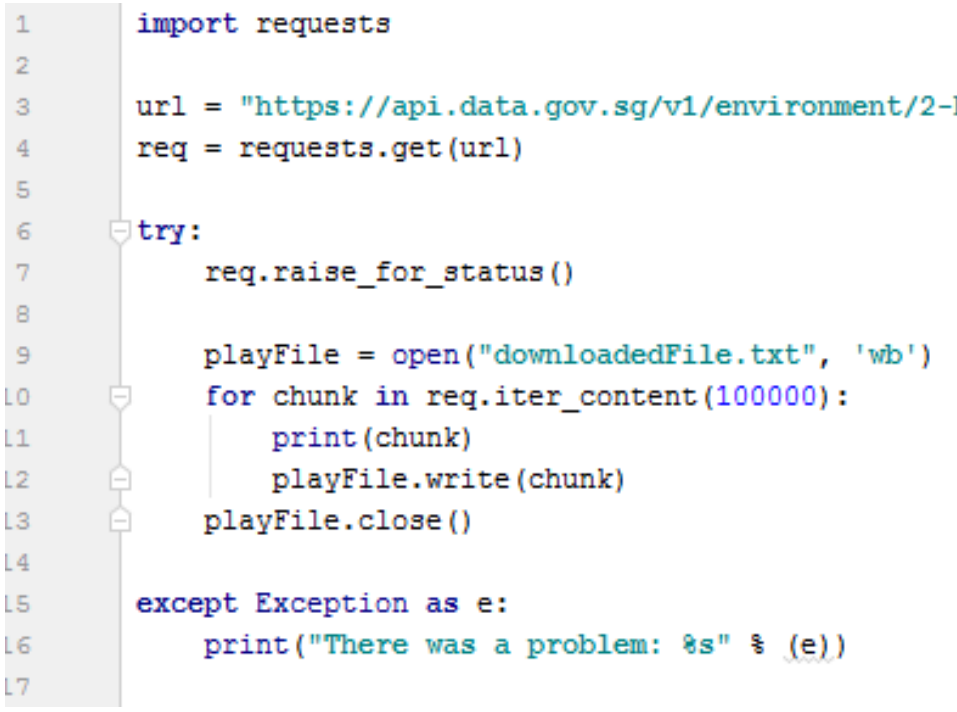
- requests – download files and web pages from the Web

Install requests module

```
import requests

url = "https://api.data.gov.sg/v1/environment/2-hour-weather-forecast"
req = requests.get(url)
print(req.text)
```

Get the required information from the given URL

Connecting to the Web

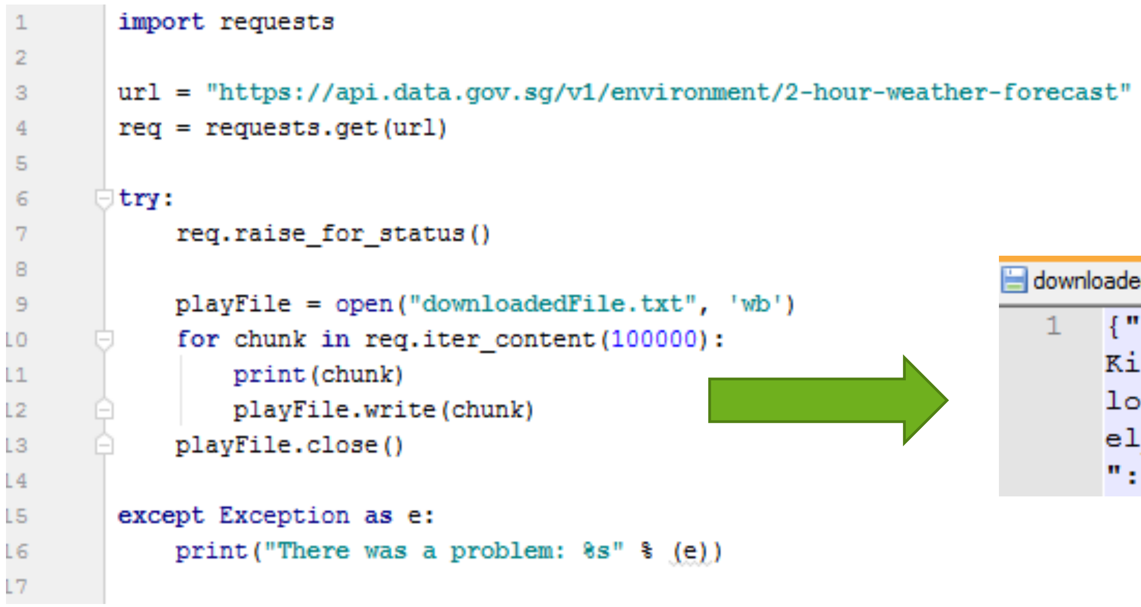


```
import requests

url = "https://api.data.gov.sg/v1/environment/2-hour-weather-forecast"
req = requests.get(url)

try:
    req.raise_for_status()

    playFile = open("downloadedFile.txt", 'wb')
    for chunk in req.iter_content(100000):
        print(chunk)
        playFile.write(chunk)
    playFile.close()

except Exception as e:
    print("There was a problem: %s" % (e))
```

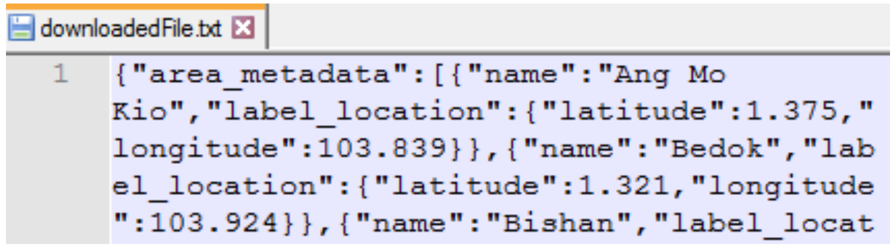
- Use requests.get() to get web content from specified URL
- Use raise_for_status() to ensure that download is successful before we continue
- Call open() with "wb" to create a new file in write binary mode
- Loop over the Response object using iter_content()
- Call write() on each iteration to write the content to the file
- Remember to close the file

Connecting to the Web

- File will be saved in "downloadedFile.txt“ (in the same folder as your program)



```python
import requests

url = "https://api.data.gov.sg/v1/environment/2-hour-weather-forecast"
req = requests.get(url)

try:
    req.raise_for_status()

    playFile = open("downloadedFile.txt", 'wb')
    for chunk in req.iter_content(100000):
        print(chunk)
        playFile.write(chunk)
    playFile.close()

except Exception as e:
    print("There was a problem: %s" % (e))
```

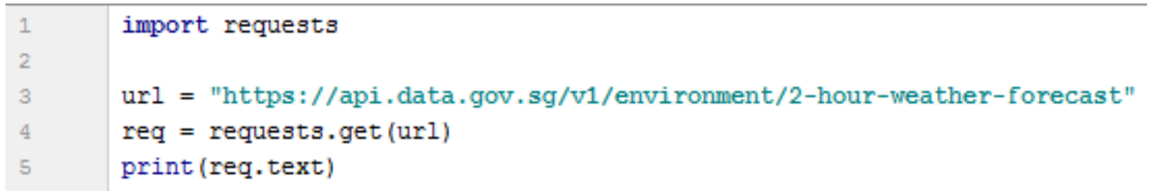


downloadedFile.txt

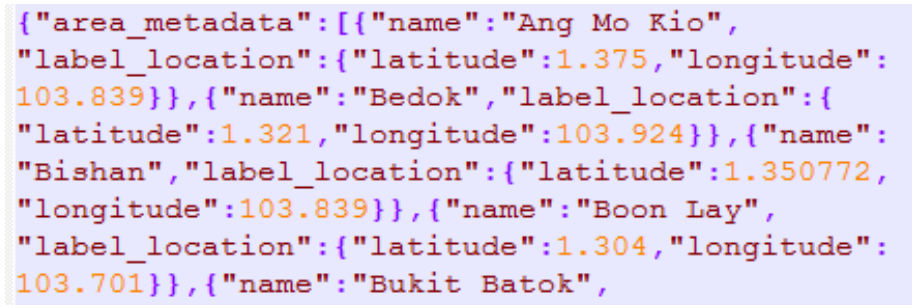
```
{"area_metadata":[{"name":"Ang Mo
Kio","label_location":{"latitude":1.375,"
longitude":103.839}},{"name":"Bedok","lab
el_location":{"latitude":1.321,"longitude
":103.924}},{"name":"Bishan","label_locat
```

Connecting to the Web

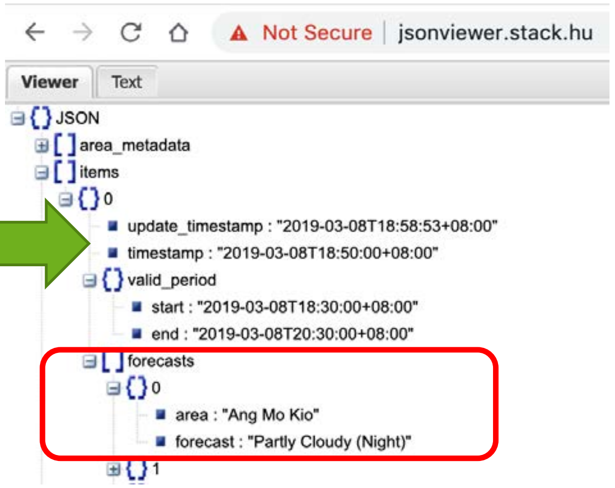
- Data is in JSON format
- Use a JSON formatter tool to present the data in a nicer form



```python
import requests

url = "https://api.data.gov.sg/v1/environment/2-hour-weather-forecast"
req = requests.get(url)
print(req.text)
```



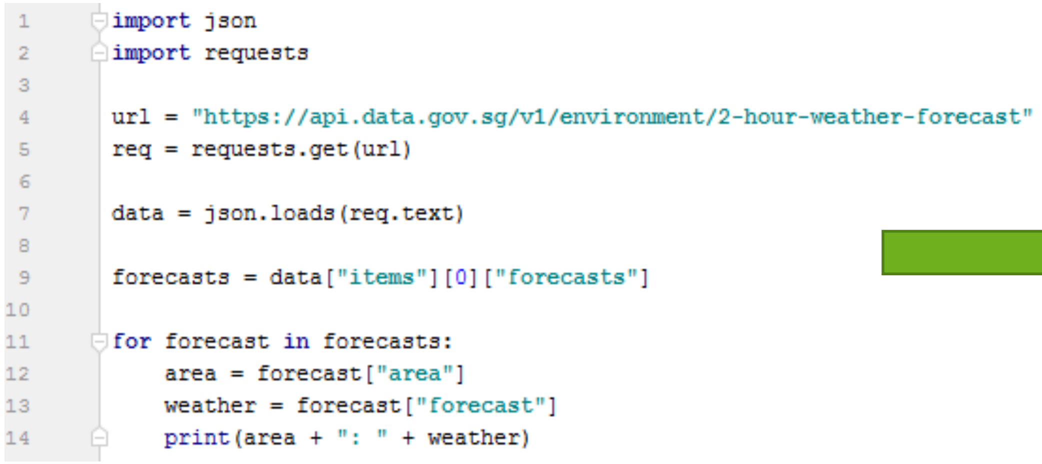
```
{"area_metadata":[{"name":"Ang Mo Kio",
"label_location":{"latitude":1.375,"longitude":
103.839}},{"name":"Bedok","label_location":{
"latitude":1.321,"longitude":103.924}},{"name":
"Bishan","label_location":{"latitude":1.350772,
"longitude":103.839}},{"name":"Boon Lay",
"label_location":{"latitude":1.304,"longitude":
103.701}},{"name":"Bukit Batok",
```



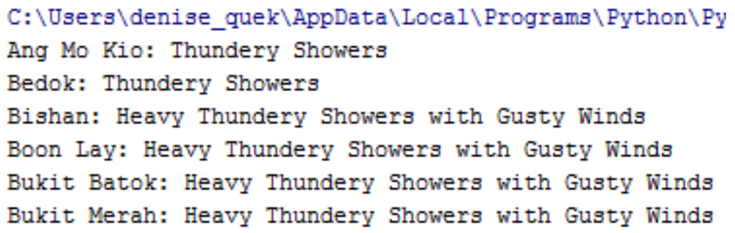
```
{} JSON
  [] area_metadata
  [] items
    {} 0
      update_timestamp : "2019-03-08T18:58:53+08:00"
      timestamp : "2019-03-08T18:50:00+08:00"
      {} valid_period
        start : "2019-03-08T18:30:00+08:00"
        end : "2019-03-08T20:30:00+08:00"
      [] forecasts
        {} 0
          area : "Ang Mo Kio"
          forecast : "Partly Cloudy (Night)"
        {} 1
```

Connecting to the Web

- To work with JSON data, import json first
- Use json.loads() to load the data in JSON format
- Extract and retrieve the required data



```python
import json
import requests

url = "https://api.data.gov.sg/v1/environment/2-hour-weather-forecast"
req = requests.get(url)

data = json.loads(req.text)

forecasts = data["items"][0]["forecasts"]

for forecast in forecasts:
    area = forecast["area"]
    weather = forecast["forecast"]
    print(area + ": " + weather)
```



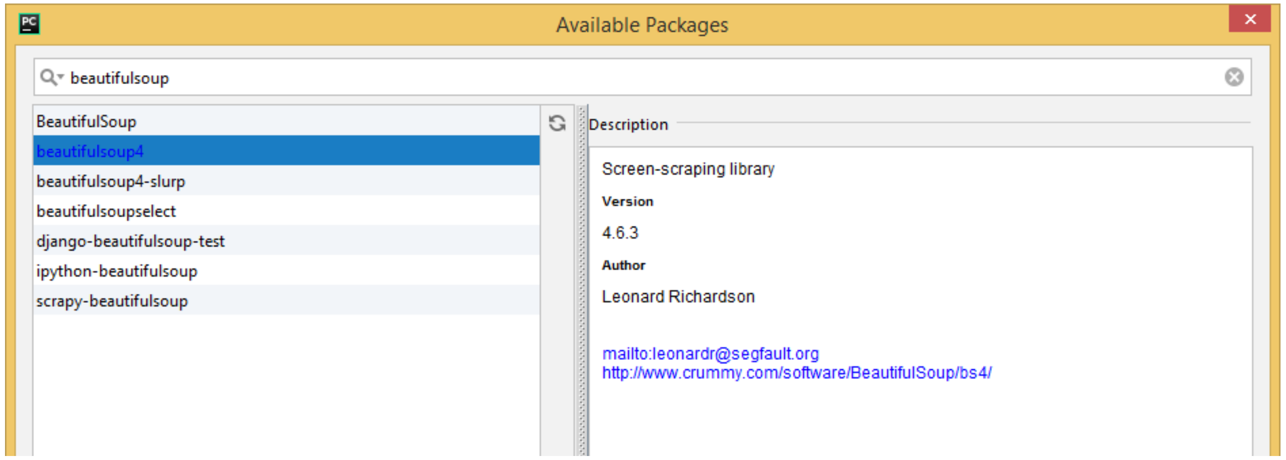
```
C:\Users\denise_quek\AppData\Local\Programs\Python\Py
Ang Mo Kio: Thundery Showers
Bedok: Thundery Showers
Bishan: Heavy Thundery Showers with Gusty Winds
Boon Lay: Heavy Thundery Showers with Gusty Winds
Bukit Batok: Heavy Thundery Showers with Gusty Winds
Bukit Merah: Heavy Thundery Showers with Gusty Winds
```

Connecting to the Web

- Beautiful Soup – a third party module that parses HTML (web pages)

Web Scraping – download and process Web content

- Install Beautiful Soup 4 -



BeautifulSoup
beautifulsoup4
beautifulsoup4-slurp
beautifulsoupselect
django-beautifulsoup-test
ipython-beautifulsoup
scrapy-beautifulsoup

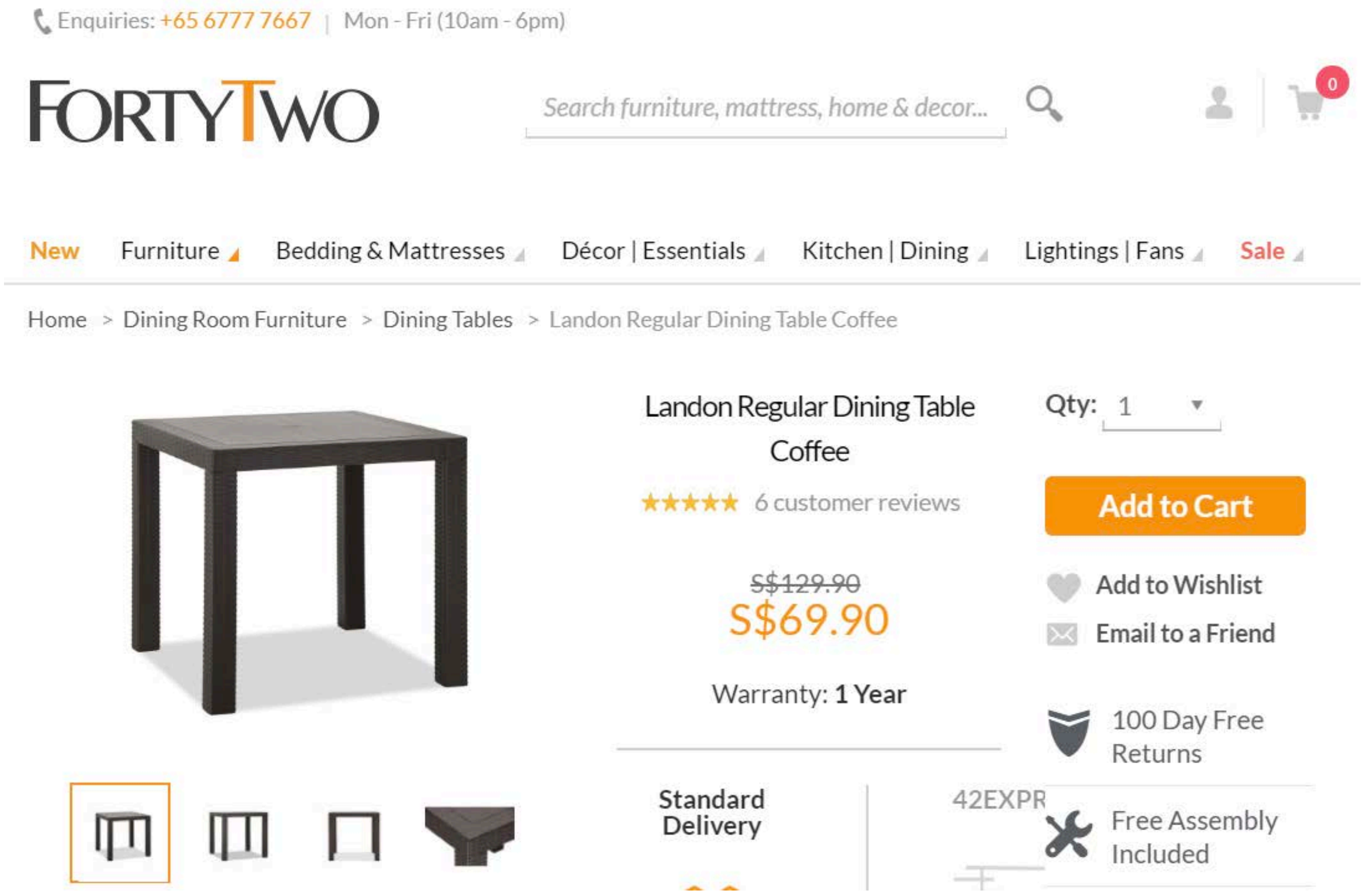
Screen-scraping library
Version
4.6.3
Author
Leonard Richardson

mailto:leonardr@segfault.org
http://www.crummy.com/software/BeautifulSoup/bs4/

Connecting to the Web

- What's the URL?

https://www.fortytwo.sg/dining/dining-tables/landon-regular-dining-table-coffee.html



Enquiries: +65 6777 7667 | Mon - Fri (10am - 6pm)

FORTYTWO

Search furniture, mattress, home & decor...

Home > Dining Room Furniture > Dining Tables > Landon Regular Dining Table Coffee

Landon Regular Dining Table Coffee

★★★★★ 6 customer reviews

~~S$129.90~~

S$69.90

Warranty: 1 Year

Qty: 1

100 Day Free Returns

Free Assembly Included

Standard Delivery

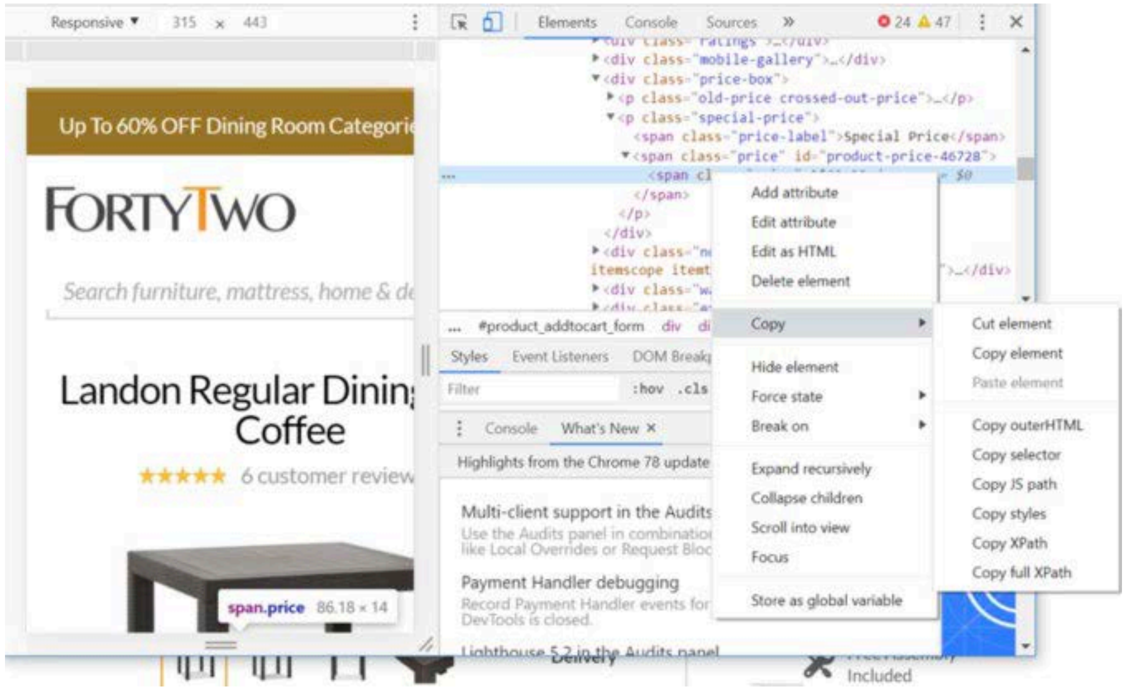
42EXPR

Connecting to the Web

- Get the url

https://www.fortytwo.sg/dining/dining-tables/ross-dining-table-walnut.html

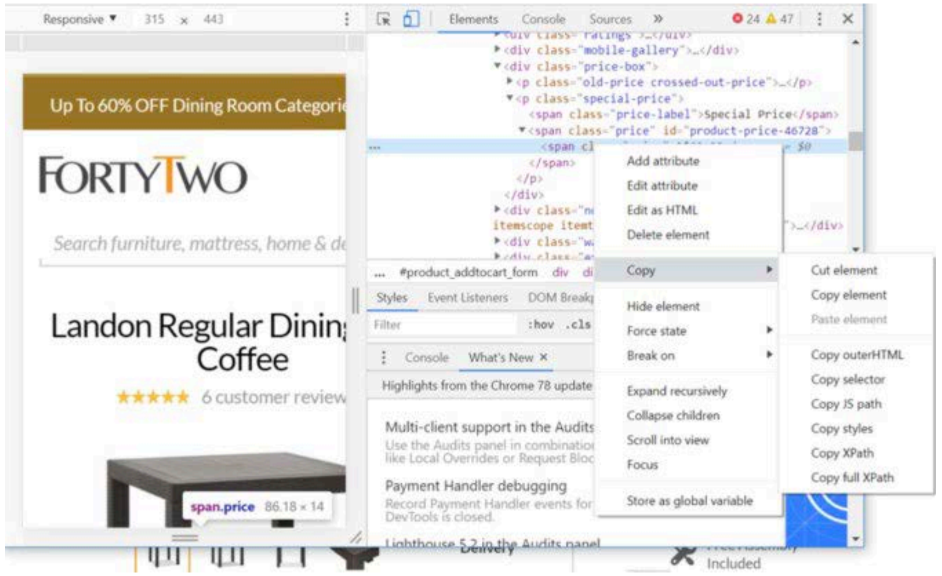
- Select the element to extract, right-click "Inspect"
- Right-click "Copy" → "Copy selector



```
▸<div class="mobile-gallery">…</div>
▾<div class="price-box">
  ▸<p class="old-price crossed-out-price">…</p>
  ▾<p class="special-price">
     <span class="price-label">Special Price</span>
    ▾<span class="price" id="product-price-46728">
       <span cl
     </span>
   </p>
 </div>
▸<div class="n
itemscope itemt
▸<div class="w
```

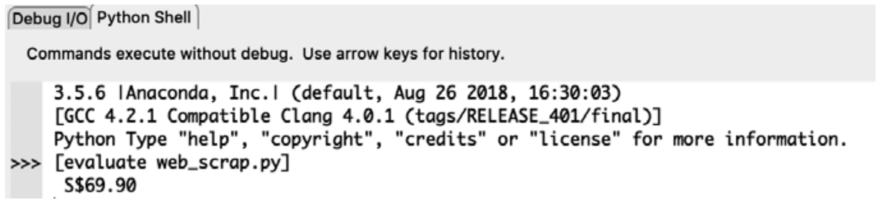
Connecting to the Web

- Get the url
- Select the element to extract, right-click "Inspect"
- Right-click "Copy" → "Copy selector"



```
<div class="mobile-gallery">…</div>
<div class="price-box">
  <p class="old-price crossed-out-price">…</p>
  <p class="special-price">
    <span class="price-label">Special Price</span>
    <span class="price" id="product-price-46728">
      <span cl
    </span>
  </p>
</div>
#product_addtocart_form div di
```

```python
import bs4
import requests

requestObj = requests.get("https://www.fortytwo.sg/dining/dining-tables/landon-regular-dining-table-coffee.html")
requestObj.raise_for_status()
soup = bs4.BeautifulSoup(requestObj.text, 'html.parser')
elements = soup.select("#product-price-46728")
print(elements[0].text)
```

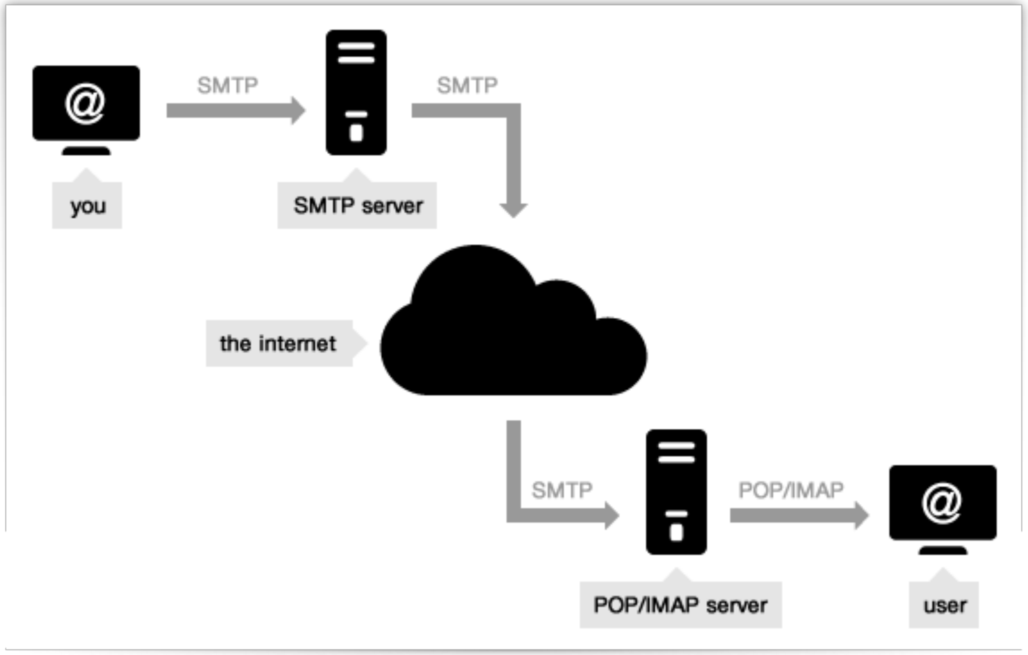


Debug I/O | Python Shell

Commands execute without debug. Use arrow keys for history.

```
    3.5.6 |Anaconda, Inc.| (default, Aug 26 2018, 16:30:03)
    [GCC 4.2.1 Compatible Clang 4.0.1 (tags/RELEASE_401/final)]
    Python Type "help", "copyright", "credits" or "license" for more information.
>>> [evaluate web_scrap.py]
     S$69.90
```

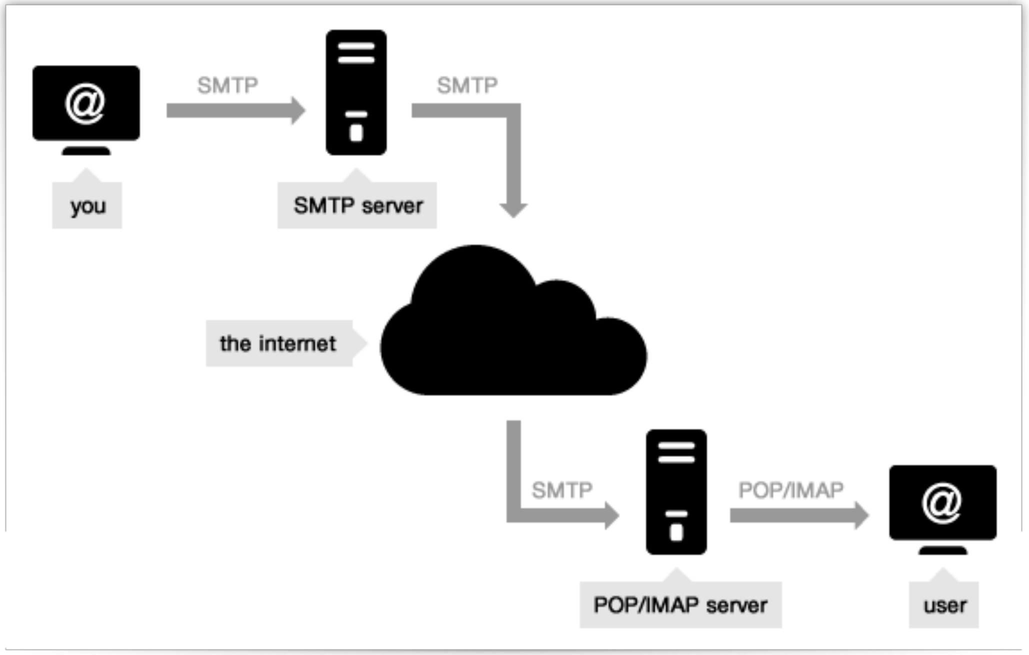
Send Email



- SMTP (Simple Mail Transfer Protocol) is used for sending and delivering from a client to a server via port 25: it's the **outgoing server**.
- IMAP and POP are two methods to access email. IMAP is the recommended method when you need to check your emails from several different devices, such as a phone, laptop, and tablet.

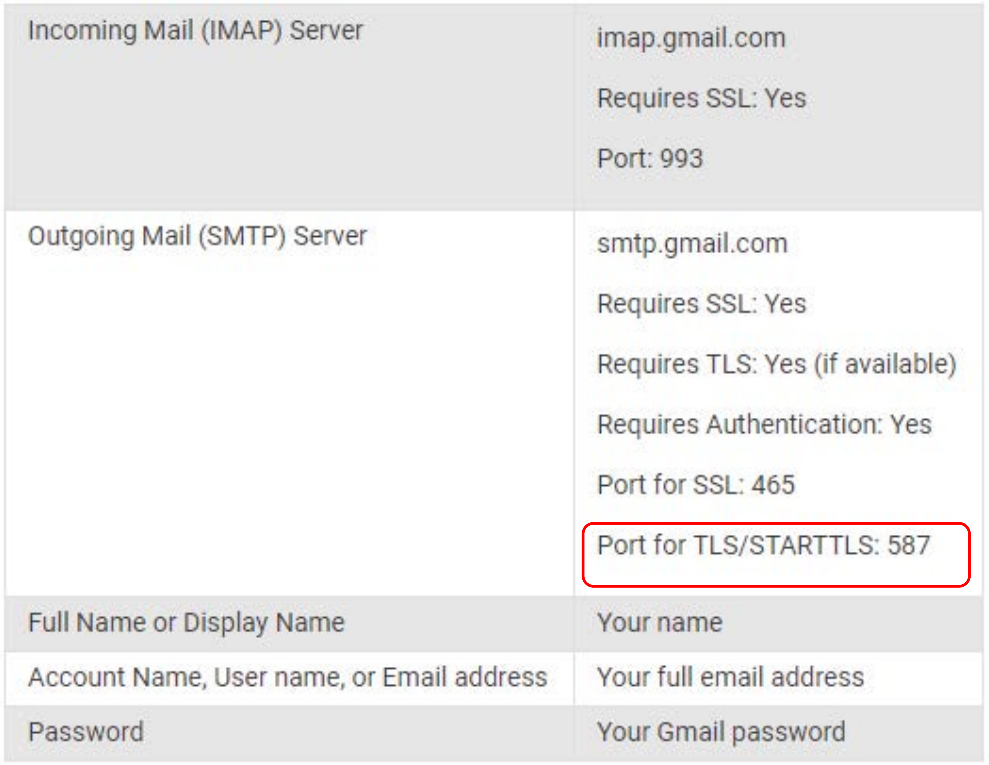
https://serversmtp.com/what-is-smtp-server/

Send Email



- **Note**: The SMTP servers used when you send your emails- Hotmail, Gmail , Yahoo Mail – are **shared among users**
- Common providers establish some **strict limits** on the number of emails you can send (e.g. Yahoo's restriction is 100 emails per hour).
- If you plan to send a bulk email or set up an email campaign you should opt for a professional outgoing email server like turboSMTP,
- which guarantees a controlled IP and ensure that all your messages reach their destination.

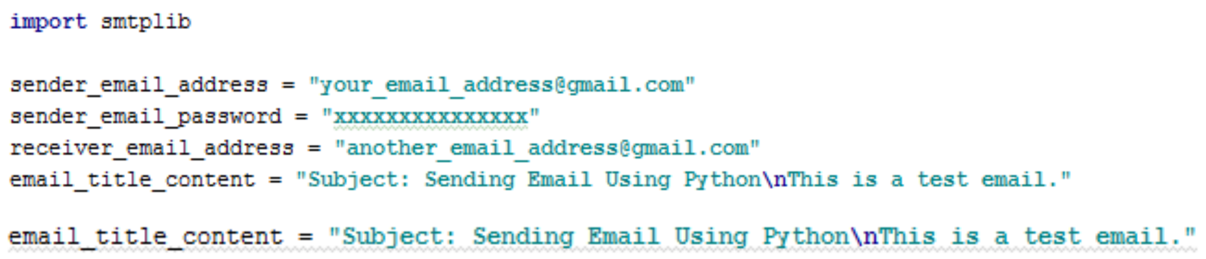
Send Email using Gmail



Incoming Mail (IMAP) Server	imap.gmail.com Requires SSL: Yes Port: 993
Outgoing Mail (SMTP) Server	smtp.gmail.com Requires SSL: Yes Requires TLS: Yes (if available) Requires Authentication: Yes Port for SSL: 465 Port for TLS/STARTTLS: 587
Full Name or Display Name	Your name
Account Name, User name, or Email address	Your full email address
Password	Your Gmail password

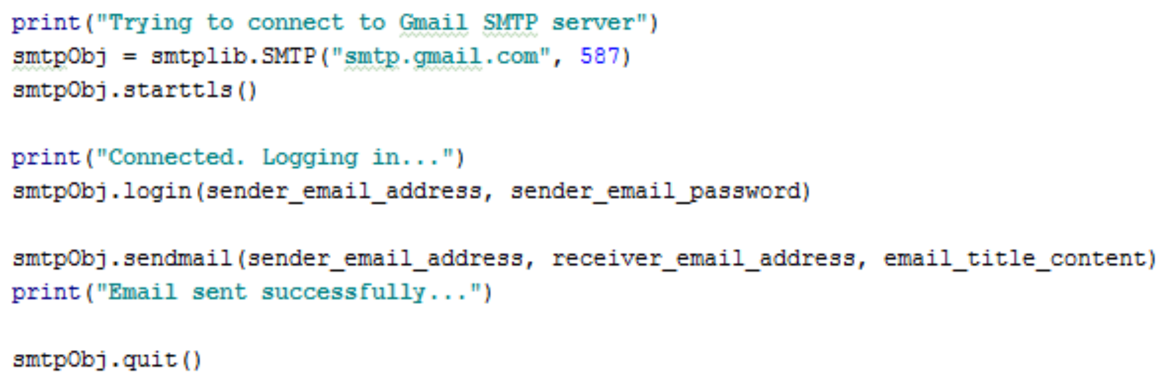
Send Email using Gmail

- Import smtplib module
- Specify Gmail email & password, receiver's email address, email title & content
- Connect to SMTP server using Port 587
- Call starttls() to enable encryption for your connection
- Login using email and password
- Call sendmail()
- Call quit() to disconnect from the SMTP server



```
import smtplib

sender_email_address = "your_email_address@gmail.com"
sender_email_password = "xxxxxxxxxxxxxxx"
receiver_email_address = "another_email_address@gmail.com"
email_title_content = "Subject: Sending Email Using Python\nThis is a test email."

email_title_content = "Subject: Sending Email Using Python\nThis is a test email."
```

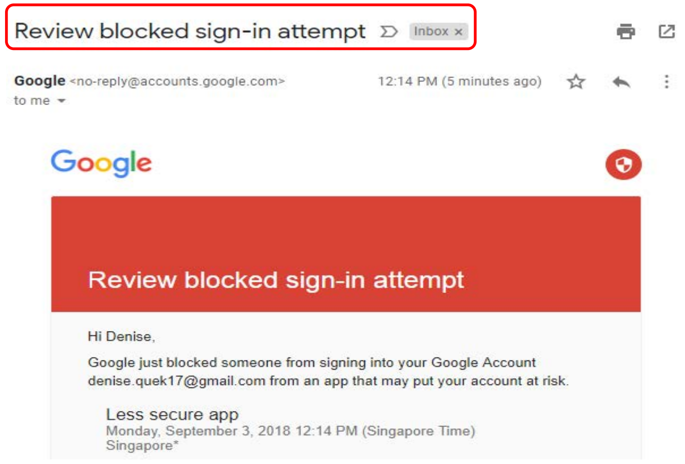
➢The start of the email body must begin with "Subject: " for the subject line. The "\n" newline character separates the subject line from the main body content.



```
print("Trying to connect to Gmail SMTP server")
smtpObj = smtplib.SMTP("smtp.gmail.com", 587)
smtpObj.starttls()

print("Connected. Logging in...")
smtpObj.login(sender_email_address, sender_email_password)

smtpObj.sendmail(sender_email_address, receiver_email_address, email_title_content)
print("Email sent successfully...")

smtpObj.quit()
```

Send Email using Gmail

- Google may block attempted sign-in from unknown devices that don't meet their security standards!



Review blocked sign-in attempt Inbox ×

Google <no-reply@accounts.google.com> 12:14 PM (5 minutes ago)
to me

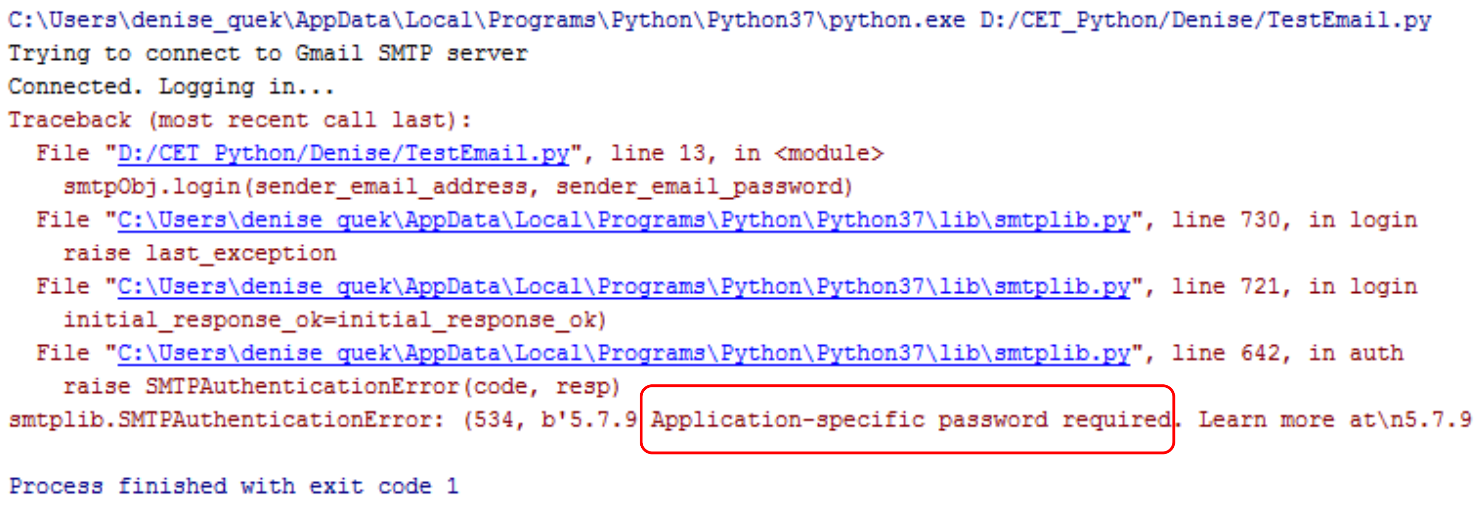
Google

Review blocked sign-in attempt

Hi Denise,

Google just blocked someone from signing into your Google Account denise.quek17@gmail.com from an app that may put your account at risk.

Less secure app
Monday, September 3, 2018 12:14 PM (Singapore Time)
Singapore*



```
C:\Users\denise_quek\AppData\Local\Programs\Python\Python37\python.exe D:/CET_Python/Denise/TestEmail.py
Trying to connect to Gmail SMTP server
Connected. Logging in...
Traceback (most recent call last):
  File "D:/CET_Python/Denise/TestEmail.py", line 13, in <module>
    smtpObj.login(sender_email_address, sender_email_password)
  File "C:\Users\denise_quek\AppData\Local\Programs\Python\Python37\lib\smtplib.py", line 730, in login
    raise last_exception
  File "C:\Users\denise_quek\AppData\Local\Programs\Python\Python37\lib\smtplib.py", line 721, in login
    initial_response_ok=initial_response_ok)
  File "C:\Users\denise_quek\AppData\Local\Programs\Python\Python37\lib\smtplib.py", line 642, in auth
    raise SMTPAuthenticationError(code, resp)
smtplib.SMTPAuthenticationError: (534, b'5.7.9 Application-specific password required. Learn more at\n5.7.9

Process finished with exit code 1
```

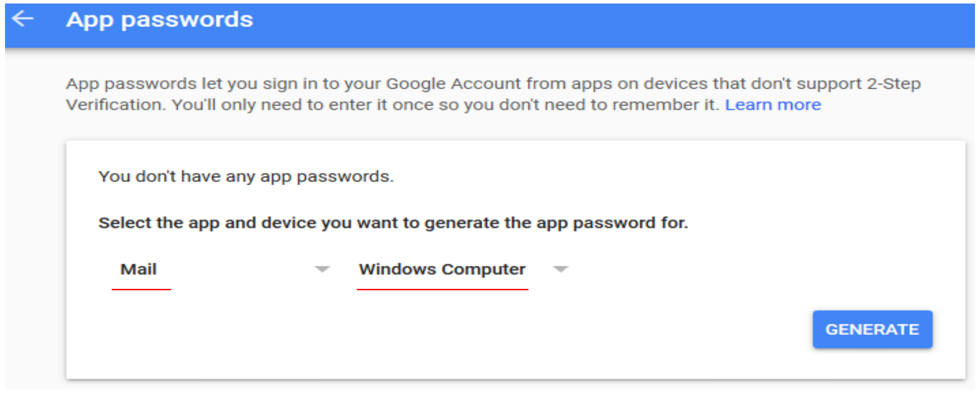
Send Email using Gmail

Steps To Create Google App Password

Step 1: Login to Gmail. Go to Account → Security, Signing in to Google

Step 2: Make sure that 2-Step Verification is on

Step 3: Create an App password



App passwords

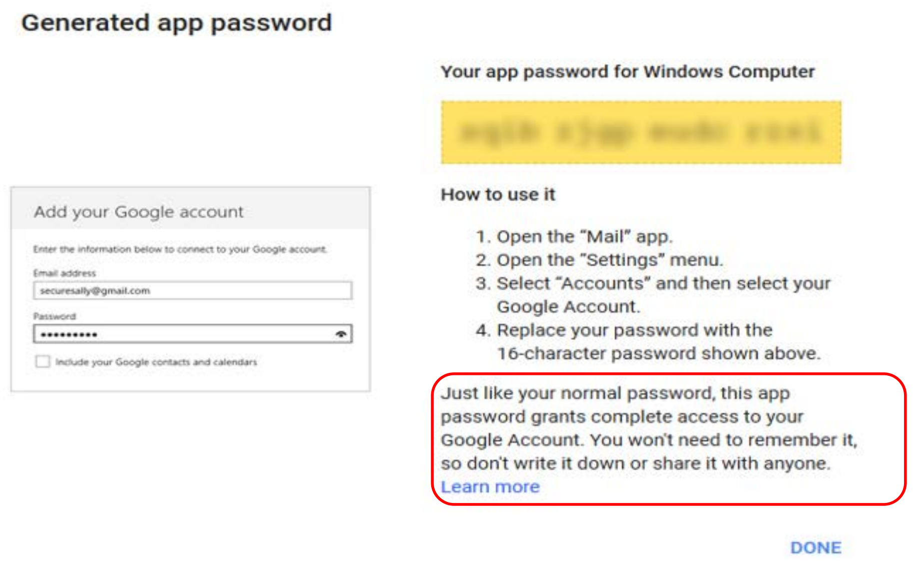
App passwords let you sign in to your Google Account from apps on devices that don't support 2-Step Verification. You'll only need to enter it once so you don't need to remember it. Learn more

You don't have any app passwords.

Select the app and device you want to generate the app password for.

Mail | Windows Computer

GENERATE



Generated app password

Add your Google account

Enter the information below to connect to your Google account.

Email address: securesally@gmail.com

Password: ••••••••••

☐ Include your Google contacts and calendars

Your app password for Windows Computer

[illegible]

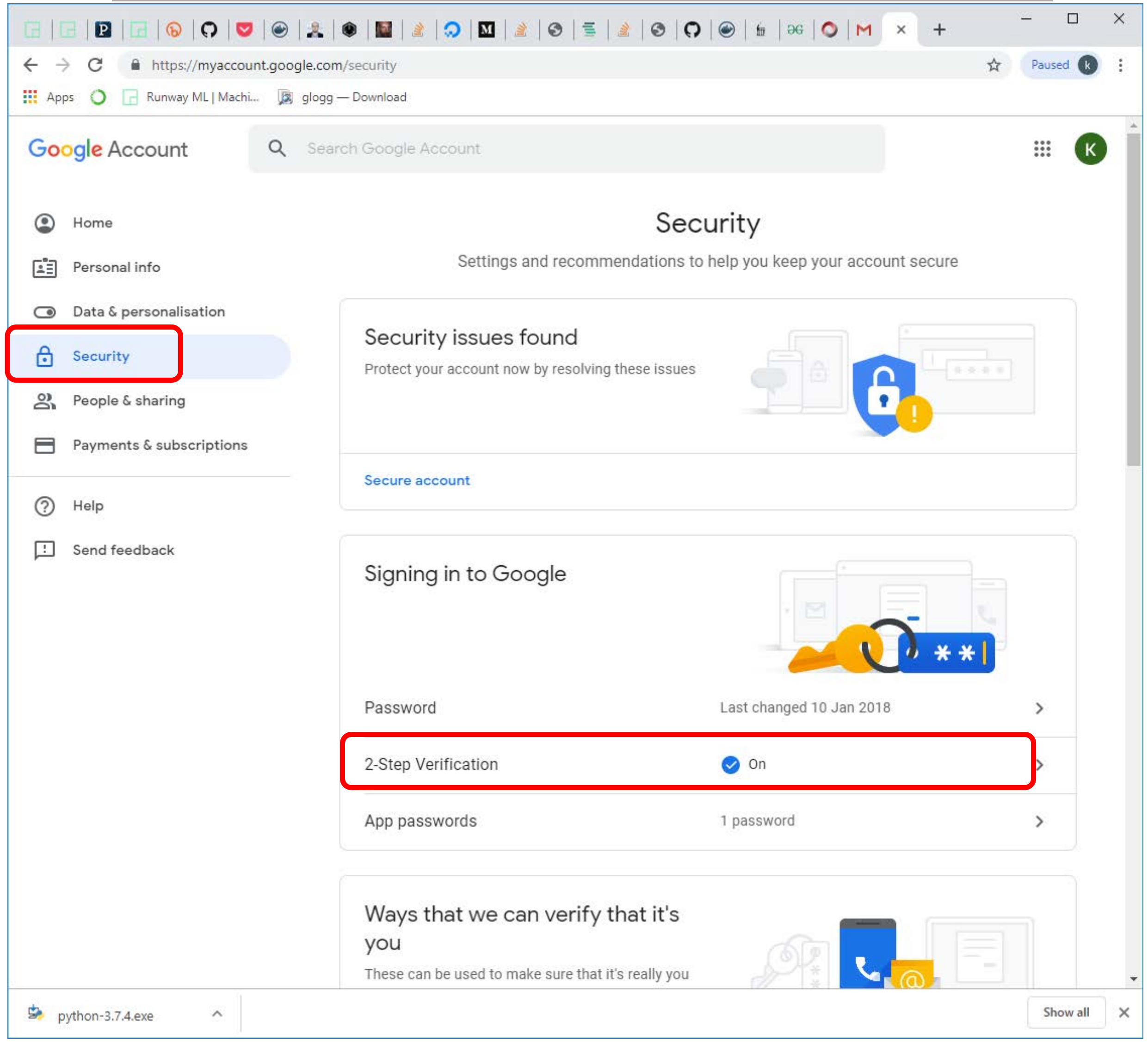
How to use it

1. Open the "Mail" app.
2. Open the "Settings" menu.
3. Select "Accounts" and then select your Google Account.
4. Replace your password with the 16-character password shown above.

Just like your normal password, this app password grants complete access to your Google Account. You won't need to remember it, so don't write it down or share it with anyone. Learn more

DONE

Send Email using Gmail



https://myaccount.google.com/security

Google Account

Search Google Account

- Home
- Personal info
- Data & personalisation
- Security
- People & sharing
- Payments & subscriptions
- Help
- Send feedback

Security

Settings and recommendations to help you keep your account secure

Security issues found

Protect your account now by resolving these issues

Secure account

Signing in to Google

Password	Last changed 10 Jan 2018
2-Step Verification	On
App passwords	1 password

Ways that we can verify that it's you

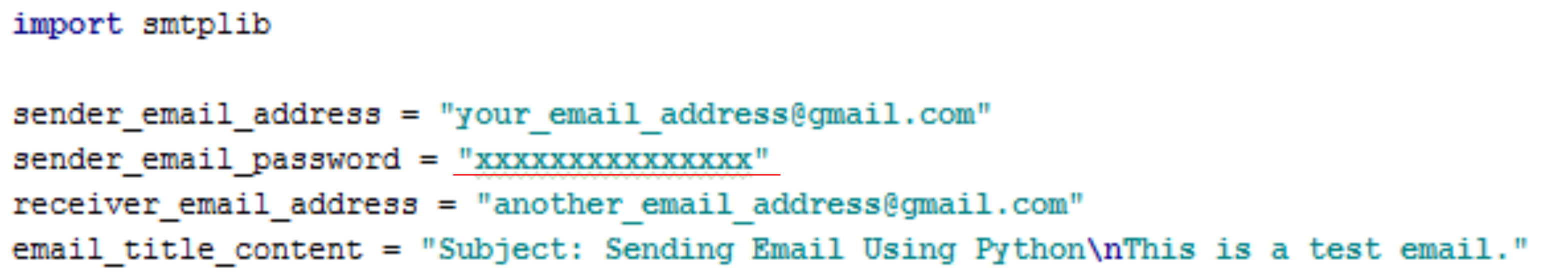
These can be used to make sure that it's really you

python-3.7.4.exe

Show all

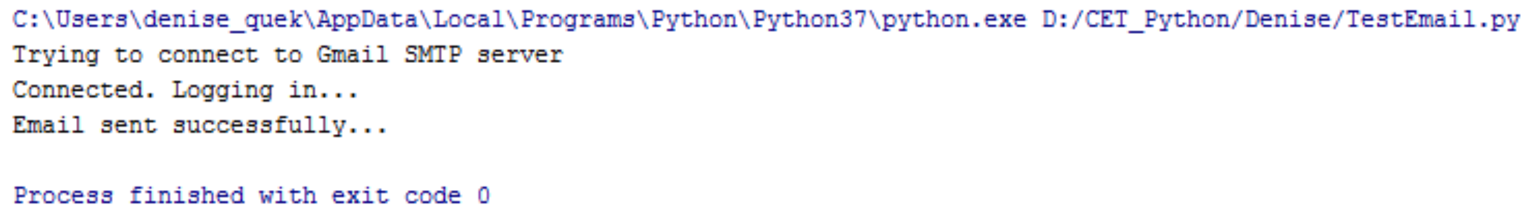
Send Email using Gmail

- Replace your actual password with the App password



```
import smtplib

sender_email_address = "your_email_address@gmail.com"
sender_email_password = "xxxxxxxxxxxxxxx"
receiver_email_address = "another_email_address@gmail.com"
email_title_content = "Subject: Sending Email Using Python\nThis is a test email."
```

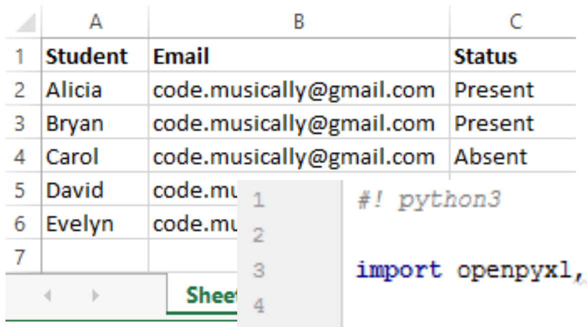
- Run your email program



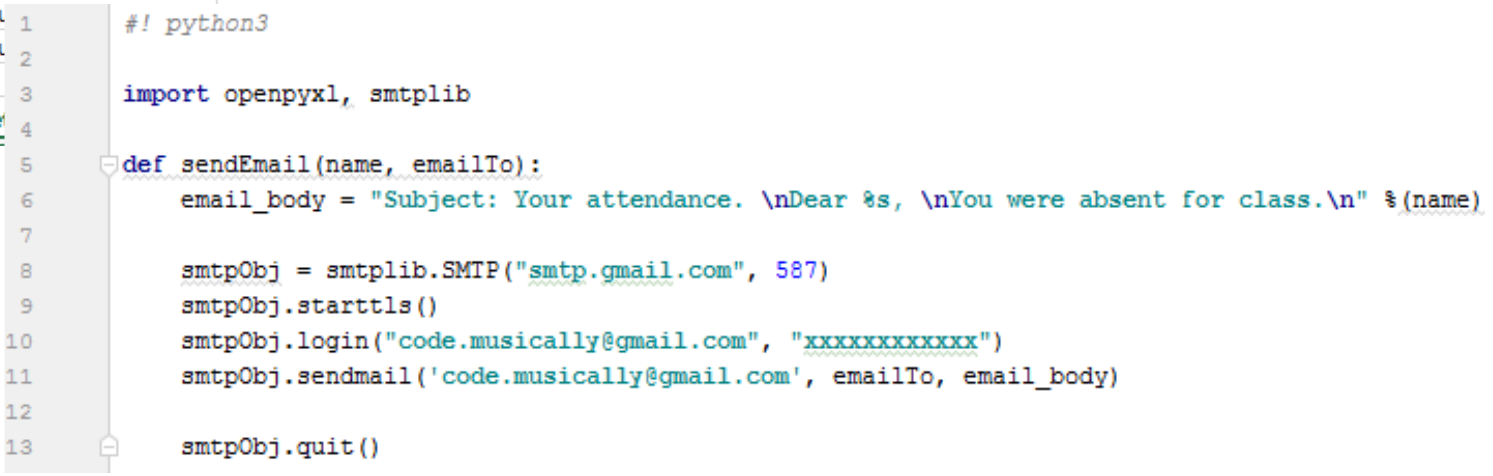
```
C:\Users\denise_quek\AppData\Local\Programs\Python\Python37\python.exe D:/CET_Python/Denise/TestEmail.py
Trying to connect to Gmail SMTP server
Connected. Logging in...
Email sent successfully...

Process finished with exit code 0
```

Send Email using Gmail

- Send email to students who were absent



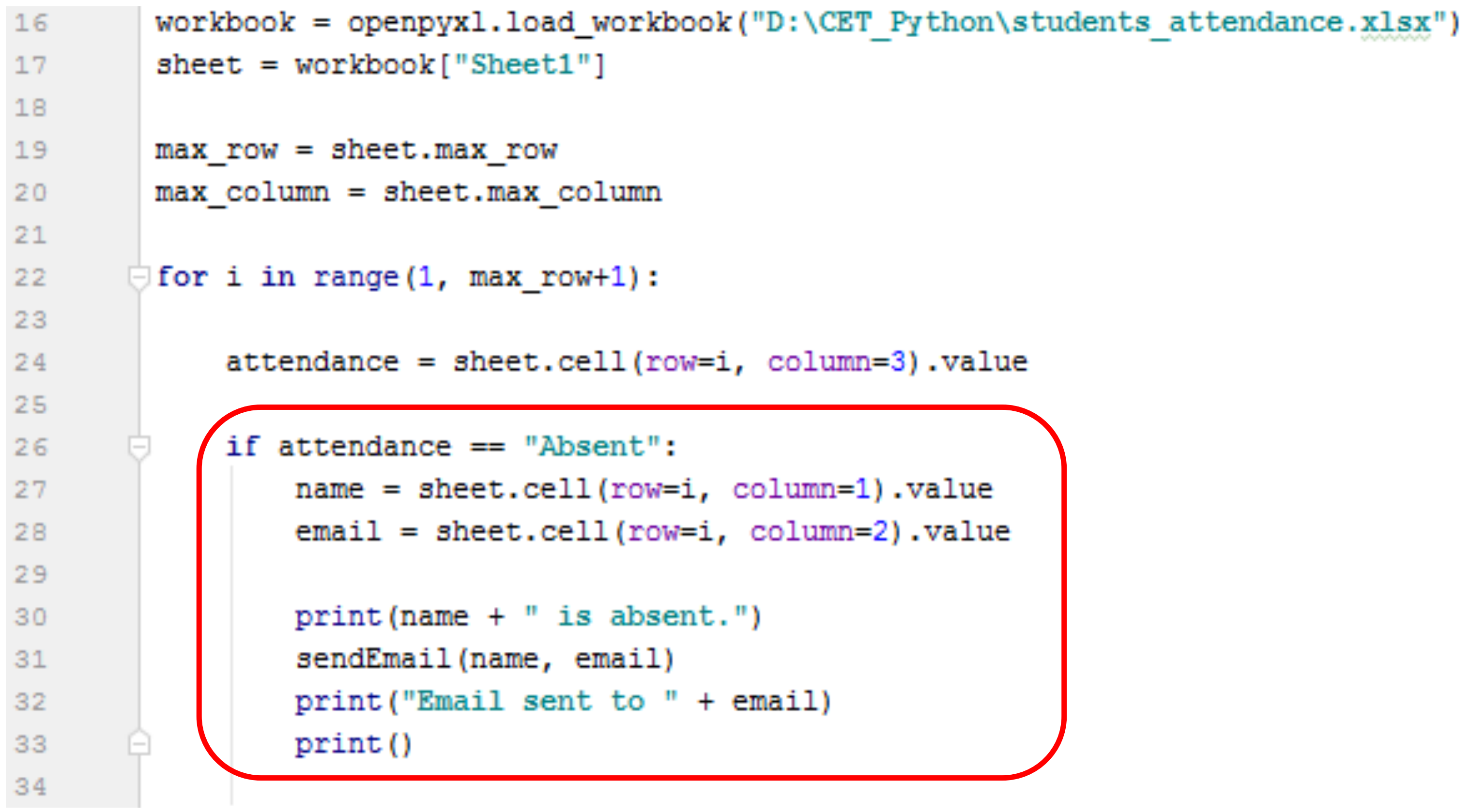
	A	B	C
1	**Student**	**Email**	**Status**
2	Alicia	code.musically@gmail.com	Present
3	Bryan	code.musically@gmail.com	Present
4	Carol	code.musically@gmail.com	Absent
5	David	code.mu	
6	Evelyn	code.mu	
7			



```
#! python3

import openpyxl, smtplib

def sendEmail(name, emailTo):
    email_body = "Subject: Your attendance. \nDear %s, \nYou were absent for class.\n" %(name)

    smtpObj = smtplib.SMTP("smtp.gmail.com", 587)
    smtpObj.starttls()
    smtpObj.login("code.musically@gmail.com", "xxxxxxxxxxxx")
    smtpObj.sendmail('code.musically@gmail.com', emailTo, email_body)

    smtpObj.quit()
```

Send Email using Gmail

- Send email to students who were absent



```
workbook = openpyxl.load_workbook("D:\CET_Python\students_attendance.xlsx")
sheet = workbook["Sheet1"]

max_row = sheet.max_row
max_column = sheet.max_column

for i in range(1, max_row+1):

    attendance = sheet.cell(row=i, column=3).value

    if attendance == "Absent":
        name = sheet.cell(row=i, column=1).value
        email = sheet.cell(row=i, column=2).value

        print(name + " is absent.")
        sendEmail(name, email)
        print("Email sent to " + email)
        print()

```

Exercise 4

- Scrap price/information from a web site and send the price/info to yourself via email

Quiz

End of Day 2

This concludes the Introduction to Python,
I hope you enjoyed it.

Thank you !

QUESTIONS ?

Where to go from here ?

Getting started step by step
    http://www.python.org/about/gettingstarted/

Run through the python tutorials:
    http://docs.python.org/tutorial/index.html

Keep the API doc under your pillow:
    http://docs.python.org/library/index.html

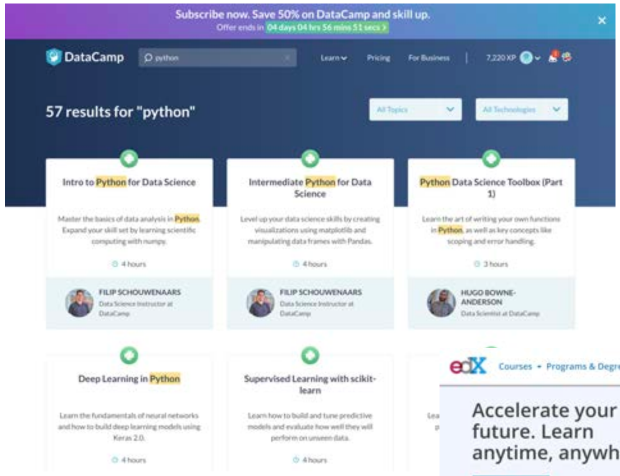
Advanced examples:
    http://www.diveintopython.org/toc/index.html

Where to go from here ?

MOOC:
DataCamp
https://www.datacamp.com/

Edx
https://www.edx.org/

Udemy (freemium course)
https://t.me/freecourse


Subscribe now. Save 50% on DataCamp and skill up.
Offer ends in 04 days 04 hrs 56 mins 51 secs >

57 results for "python"

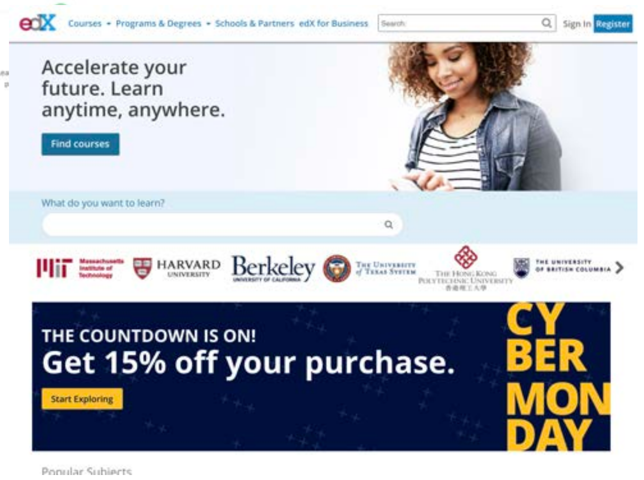
Intro to Python for Data Science
Master the basics of data analysis in Python. Expand your skill set by learning scientific computing with numpy.
4 hours
FILIP SCHOUWENAARS
Data Science Instructor at DataCamp

Intermediate Python for Data Science
Level up your data science skills by creating visualizations using matplotlib and manipulating data frames with Pandas.
4 hours
FILIP SCHOUWENAARS
Data Science Instructor at DataCamp

Python Data Science Toolbox (Part 1)
Learn the art of writing your own functions in Python, as well as key concepts like scoping and error handling.
3 hours
HUGO BOWNE-ANDERSON
Data Scientist at DataCamp

Deep Learning in Python
Learn the fundamentals of neural networks and how to build deep learning models using Keras 2.0.
4 hours

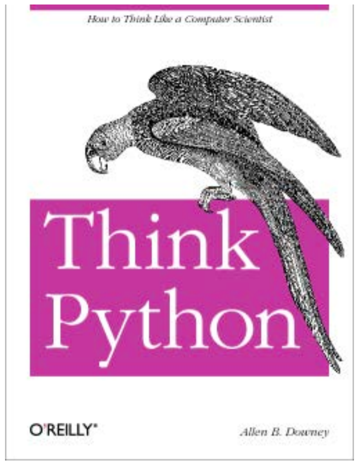
Supervised Learning with scikit-learn
Learn how to build and tune predictive models and evaluate how well they will perform on unseen data.
4 hours


Accelerate your future. Learn anytime, anywhere.

What do you want to learn?

THE COUNTDOWN IS ON!
Get 15% off your purchase.

Where to go from here ?



Think Python is an introduction to Python programming for beginners. It starts with basic concepts of programming, and is carefully designed to define all terms when they are first used and to develop each new concept in a logical progression. Larger pieces, like recursion and object-oriented programming are divided into a sequence of smaller steps and introduced over the course of several chapters.

Think Python is a Free Book. It is available under the Creative Commons Attribution-NonCommercial 3.0 Unported License, which means that you are free to copy, distribute, and modify it, as long as you attribute the work and don't use it for commercial purposes.
http://greenteapress.com/thinkpython/thinkpython.pdf

SKILLSfuture SG

Lifelong Learning

Scan me

- https://www.rp.edu.sg/soi/lifelong-learning

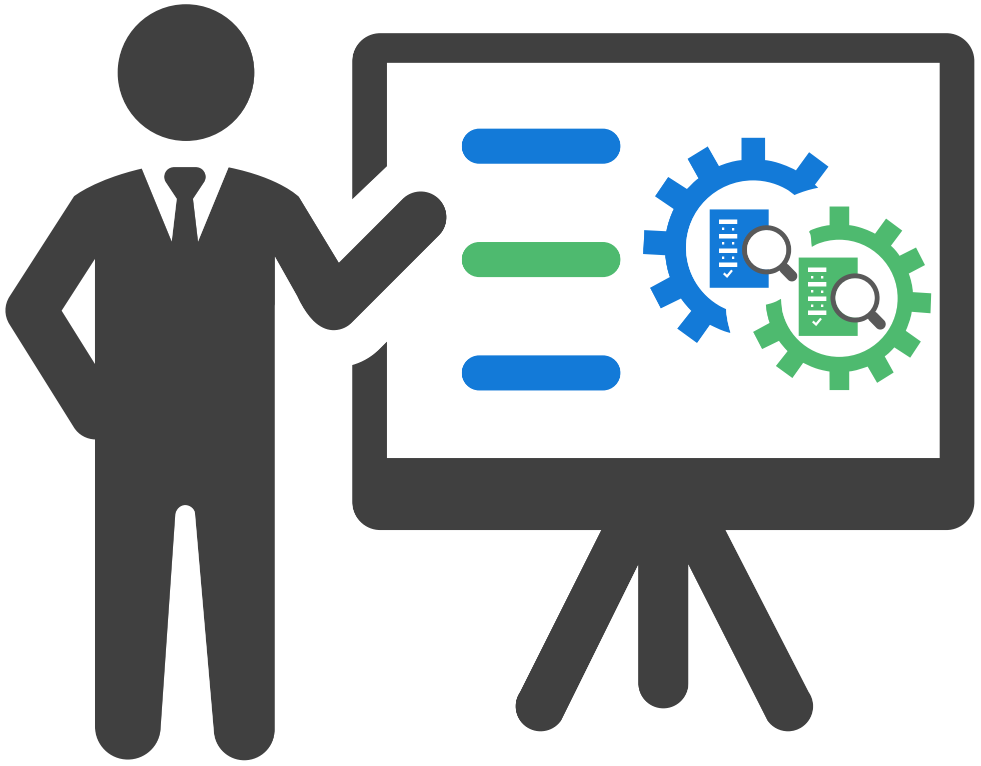
Short Courses

SOI offers an extensive variety of short, industry-relevant courses for ICT skills upgrading and skills acquisition. Our courses are categorized under different areas, ranging from Artificial Intelligence (AI), Business Intelligence/ Business Analytics (BI/BA), Business Processes (BP), Unmanned Aerial Vehicle (UAV), IT Security, New/ Digital Media, Software Development to the Internet of Things (IoT). To view our short course offerings, click on the relevant tab below.

AI | Data Analytics | IT Security | DevOps | Software Development | New/ Digital Media | UAV | RPA

+ Artificial Intelligence for Everyone - A Practical Experience (1 day Beginner)
+ Artificial Intelligence for Techies - A Hands-On Approach (1 day Beginner)
+ An Introduction to Code-Free Machine Learning (1 day Beginner)

Thank you

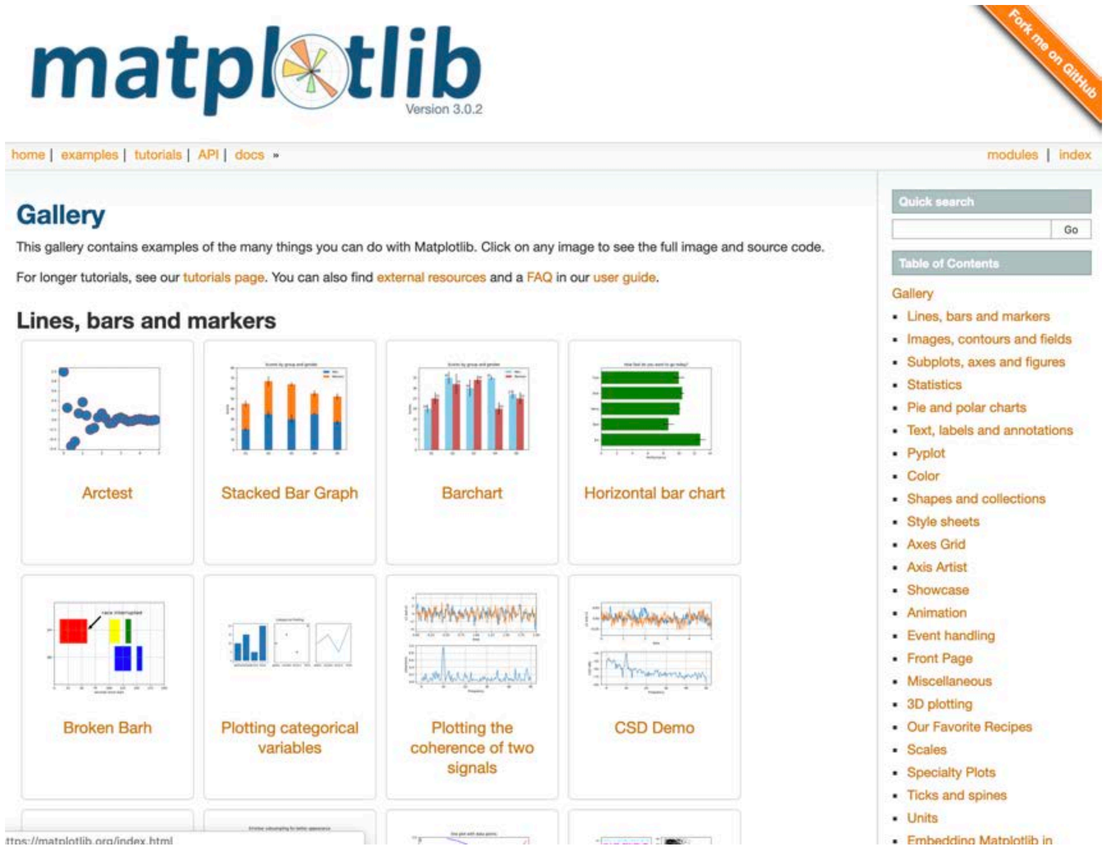


Email
seow_khee_wei@rp.edu.sg

Telegram
@kwseow

Source code: http://bit.ly/2vXKZIL

Charting



matplotlib Version 3.0.2

Fork me on GitHub

home | examples | tutorials | API | docs »

modules | index

Gallery

This gallery contains examples of the many things you can do with Matplotlib. Click on any image to see the full image and source code.

For longer tutorials, see our tutorials page. You can also find external resources and a FAQ in our user guide.

Lines, bars and markers

Arctest

Stacked Bar Graph

Barchart

Horizontal bar chart

Broken Barh

Plotting categorical variables

Plotting the coherence of two signals

CSD Demo

ttps://matplotlib.org/index.html

Quick search

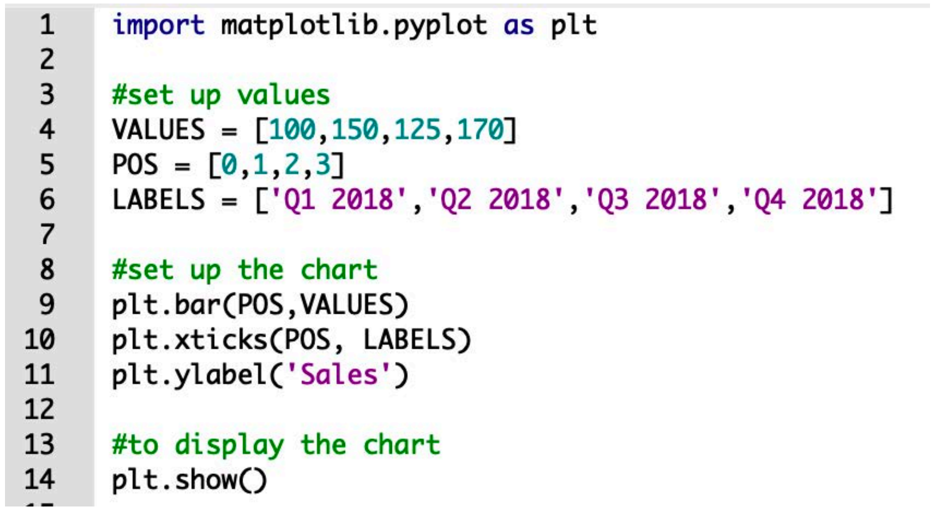
Go

Table of Contents

Install matplotlib

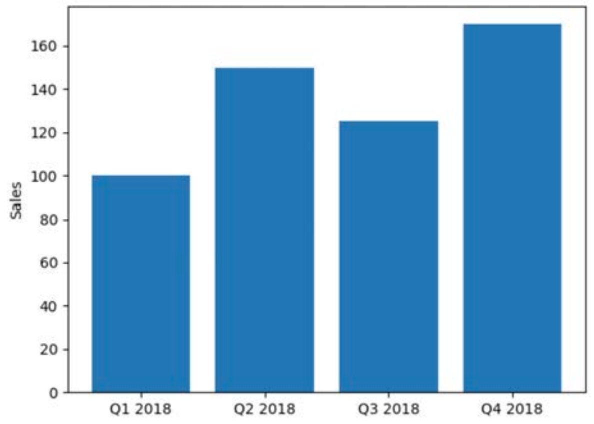
Full documentation:
https://matplotlib.org/

Charting



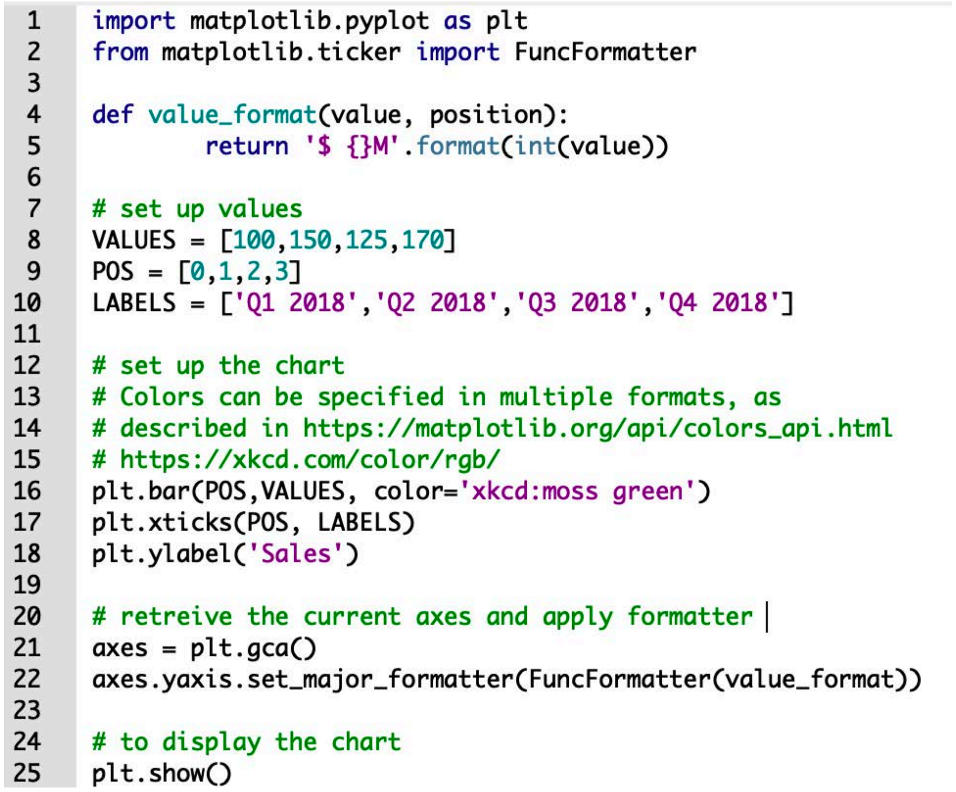
```python
import matplotlib.pyplot as plt

#set up values
VALUES = [100,150,125,170]
POS = [0,1,2,3]
LABELS = ['Q1 2018','Q2 2018','Q3 2018','Q4 2018']

#set up the chart
plt.bar(POS,VALUES)
plt.xticks(POS, LABELS)
plt.ylabel('Sales')

#to display the chart
plt.show()
```

- Install matplotlib
- Prepare data
- Create bar graph
- Display the chart



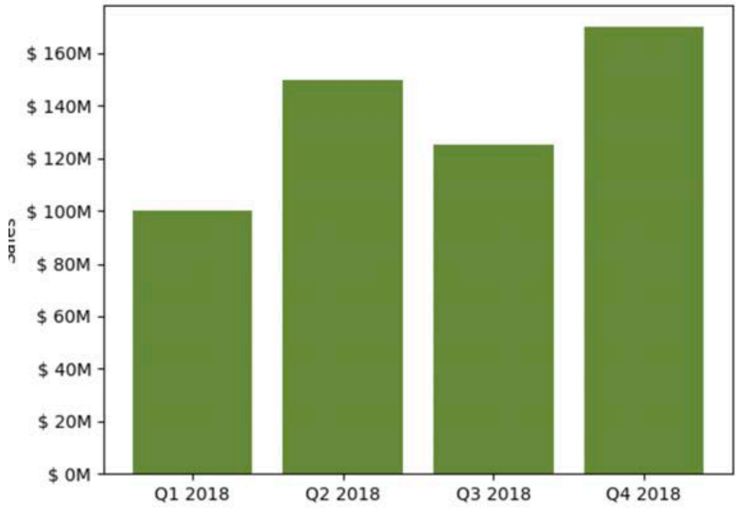
https://matplotlib.org/api/_as_gen/matplotlib.pyplot.bar.html

Charting

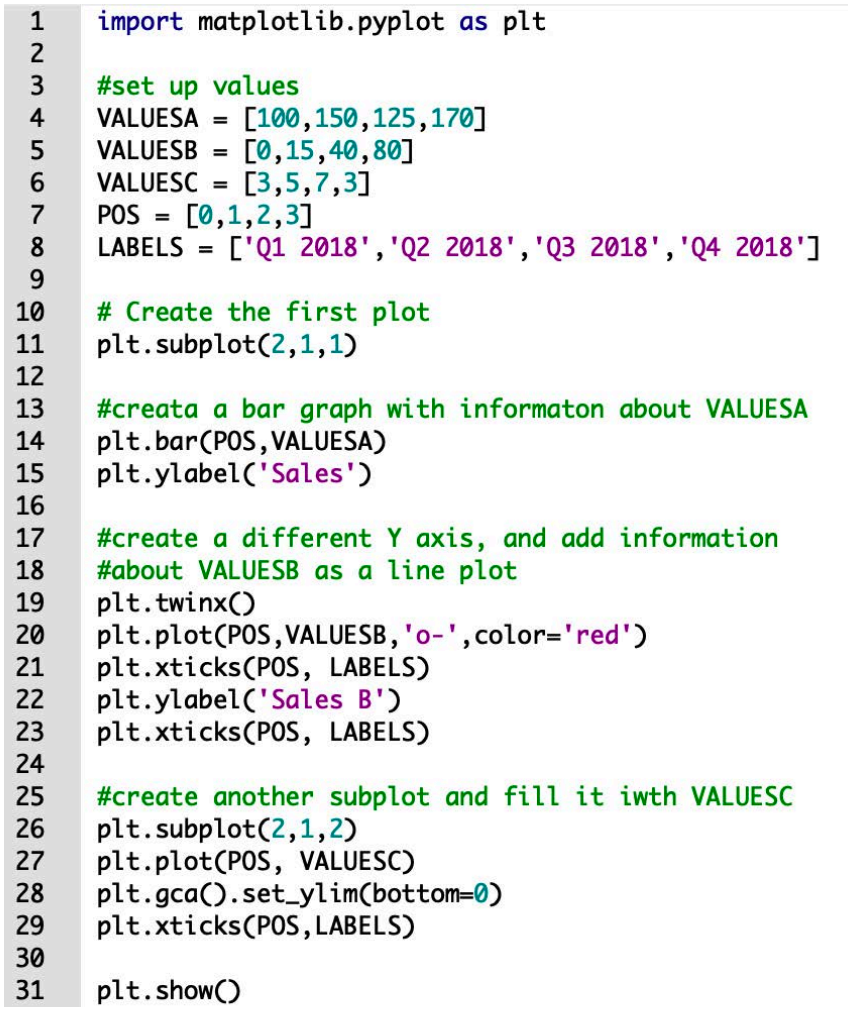


```python
import matplotlib.pyplot as plt
from matplotlib.ticker import FuncFormatter

def value_format(value, position):
        return '$ {}M'.format(int(value))

# set up values
VALUES = [100,150,125,170]
POS = [0,1,2,3]
LABELS = ['Q1 2018','Q2 2018','Q3 2018','Q4 2018']

# set up the chart
# Colors can be specified in multiple formats, as
# described in https://matplotlib.org/api/colors_api.html
# https://xkcd.com/color/rgb/
plt.bar(POS,VALUES, color='xkcd:moss green')
plt.xticks(POS, LABELS)
plt.ylabel('Sales')

# retreive the current axes and apply formatter
axes = plt.gca()
axes.yaxis.set_major_formatter(FuncFormatter(value_format))

# to display the chart
plt.show()
```

- Install matplotlib
- Prepare data
- Customise graph options
- Create bar graph
- Display the chart

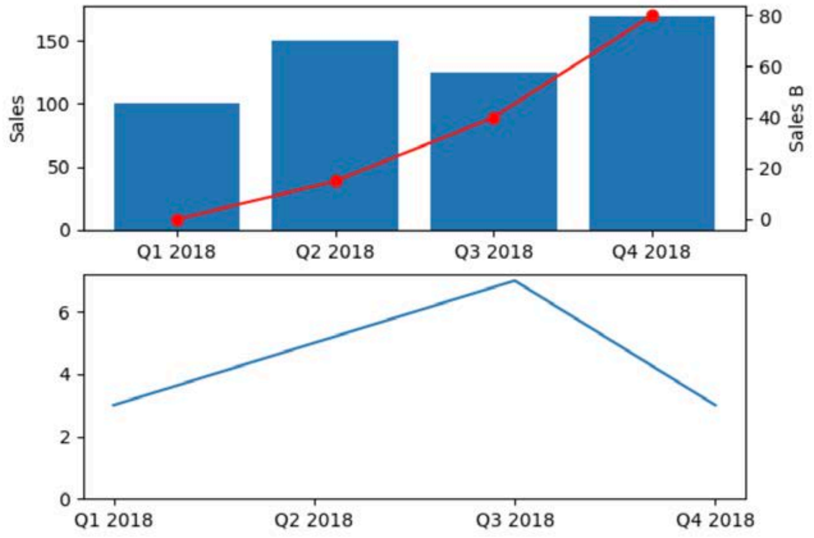


Charting



```
import matplotlib.pyplot as plt

#set up values
VALUESA = [100,150,125,170]
VALUESB = [0,15,40,80]
VALUESC = [3,5,7,3]
POS = [0,1,2,3]
LABELS = ['Q1 2018','Q2 2018','Q3 2018','Q4 2018']

# Create the first plot
plt.subplot(2,1,1)

#creata a bar graph with informaton about VALUESA
plt.bar(POS,VALUESA)
plt.ylabel('Sales')

#create a different Y axis, and add information
#about VALUESB as a line plot
plt.twinx()
plt.plot(POS,VALUESB,'o-',color='red')
plt.xticks(POS, LABELS)
plt.ylabel('Sales B')
plt.xticks(POS, LABELS)

#create another subplot and fill it iwth VALUESC
plt.subplot(2,1,2)
plt.plot(POS, VALUESC)
plt.gca().set_ylim(bottom=0)
plt.xticks(POS,LABELS)

plt.show()
```

- Multiple charts

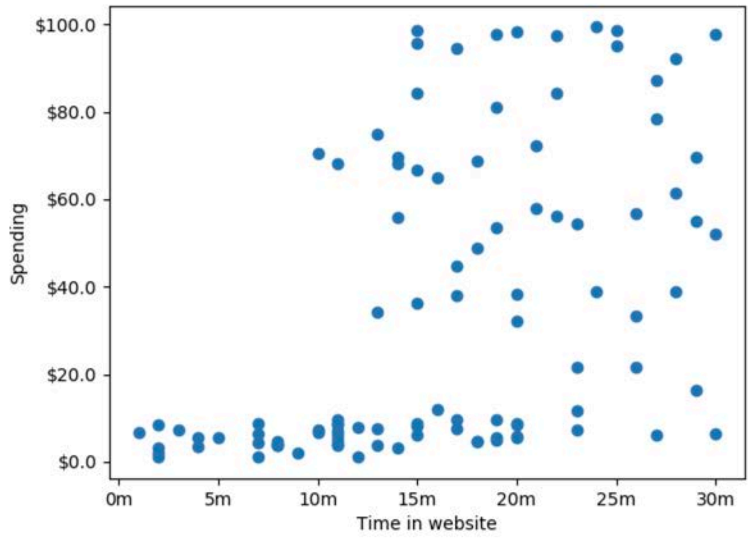


https://matplotlib.org/api/_as_gen/matplotlib.pyplot.subplot.html

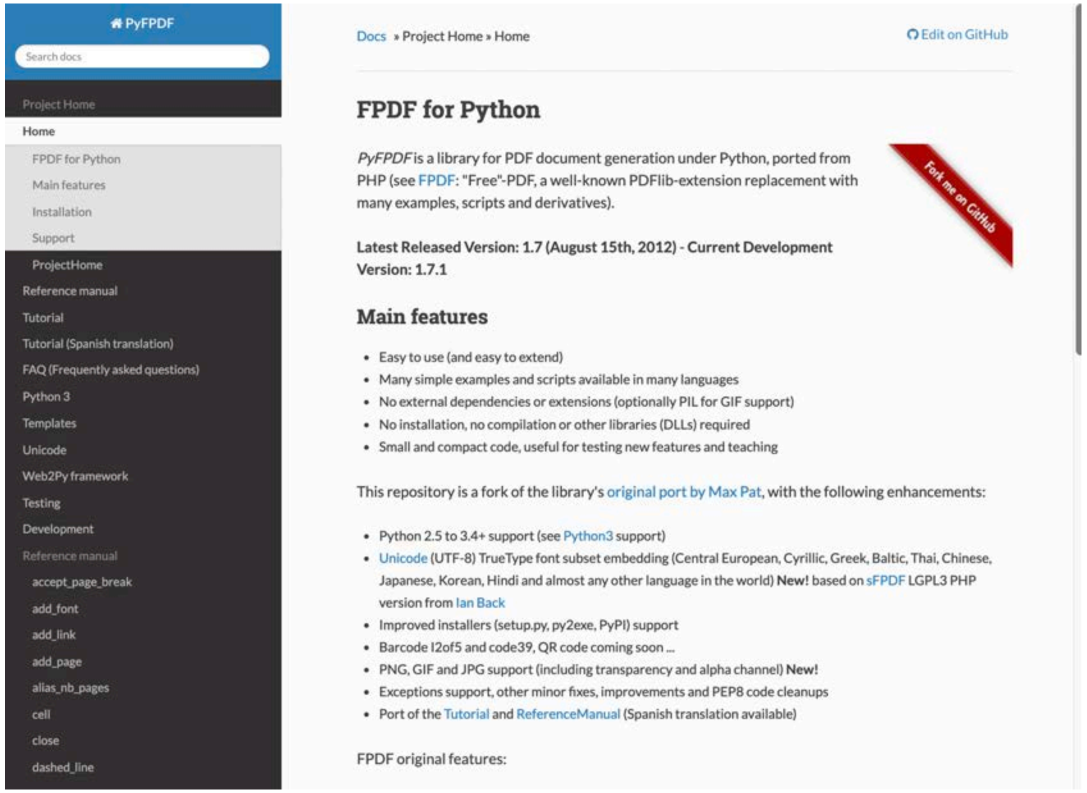
Charting – Scatter Plot

```
import csv
import matplotlib.pyplot as plt
from matplotlib.ticker import FuncFormatter

def format_minutes(value, pos):
        return '{}m'.format(int(value))

def format_dollars(value, pos):
        return '${}'.format(value)

# read data from csv
fp = open("scatter.csv","r", newline='')
reader = csv.reader(fp)
data = list(reader)

data_x=[]
data_y=[]
for x, y in data:
        data_x.append(float(x))
        data_y.append(float(y))

plt.scatter(data_x, data_y)

plt.gca().xaxis.set_major_formatter(FuncFormatter(format_minutes))
plt.xlabel('Time in website')
plt.gca().yaxis.set_major_formatter(FuncFormatter(format_dollars))
plt.ylabel('Spending')

plt.show()
```

- To save a plot: plt.savefig(*filename*)
- Save the plot before you display



PDF



FPDF for Python

PyFPDF is a library for PDF document generation under Python, ported from PHP (see FPDF: "Free"-PDF, a well-known PDFlib-extension replacement with many examples, scripts and derivatives).

Latest Released Version: 1.7 (August 15th, 2012) - Current Development Version: 1.7.1

Main features

- Easy to use (and easy to extend)
- Many simple examples and scripts available in many languages
- No external dependencies or extensions (optionally PIL for GIF support)
- No installation, no compilation or other libraries (DLLs) required
- Small and compact code, useful for testing new features and teaching

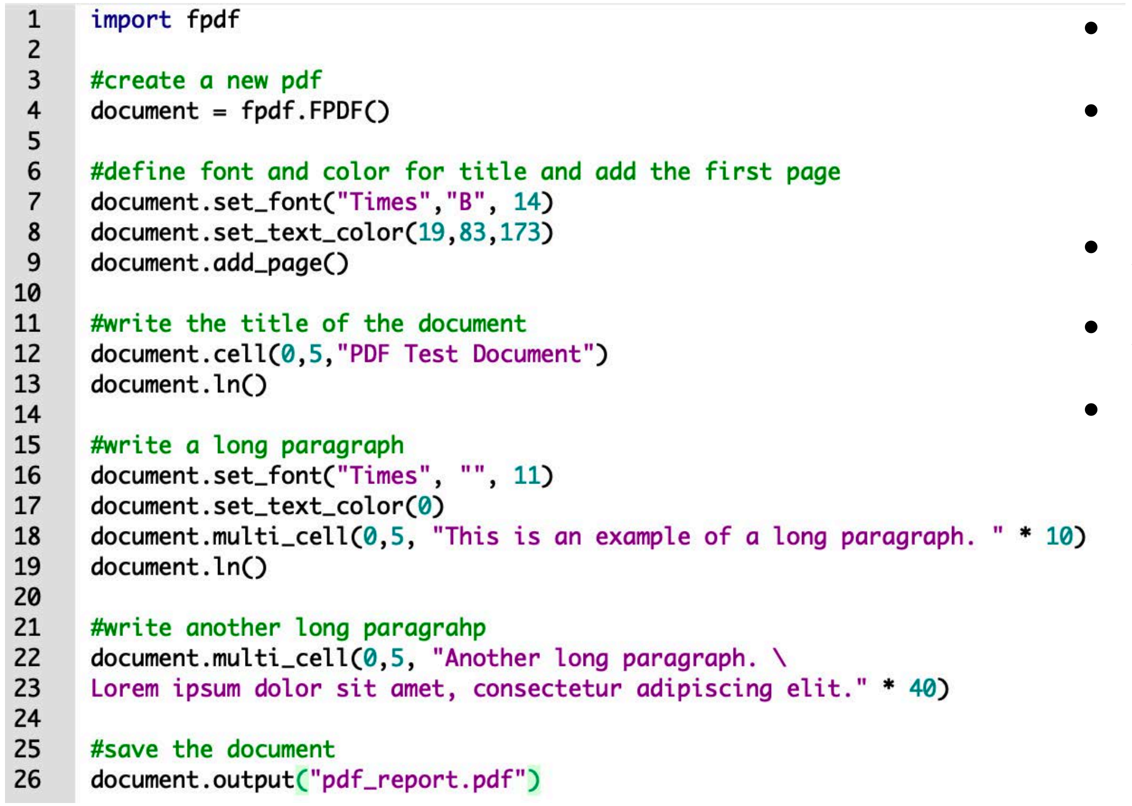
This repository is a fork of the library's original port by Max Pat, with the following enhancements:

- Python 2.5 to 3.4+ support (see Python3 support)
- Unicode (UTF-8) TrueType font subset embedding (Central European, Cyrillic, Greek, Baltic, Thai, Chinese, Japanese, Korean, Hindi and almost any other language in the world) **New!** based on sFPDF LGPL3 PHP version from Ian Back
- Improved installers (setup.py, py2exe, PyPI) support
- Barcode I2of5 and code39, QR code coming soon ...
- PNG, GIF and JPG support (including transparency and alpha channel) **New!**
- Exceptions support, other minor fixes, improvements and PEP8 code cleanups
- Port of the Tutorial and ReferenceManual (Spanish translation available)

FPDF original features:

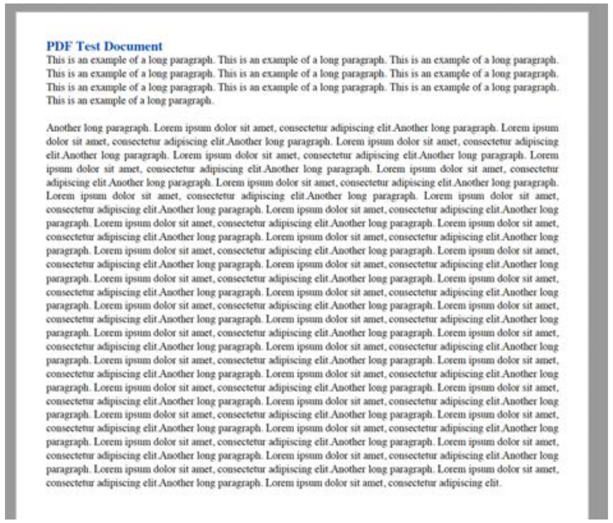
- Install fpdf
  - pip install fpdf

PDF – Basic document



```
import fpdf

#create a new pdf
document = fpdf.FPDF()

#define font and color for title and add the first page
document.set_font("Times","B", 14)
document.set_text_color(19,83,173)
document.add_page()

#write the title of the document
document.cell(0,5,"PDF Test Document")
document.ln()

#write a long paragraph
document.set_font("Times", "", 11)
document.set_text_color(0)
document.multi_cell(0,5, "This is an example of a long paragraph. " * 10)
document.ln()

#write another long paragrahp
document.multi_cell(0,5, "Another long paragraph. \
Lorem ipsum dolor sit amet, consectetur adipiscing elit." * 40)

#save the document
document.output("pdf_report.pdf")
```

- Import fpdf
- Create a new pdf document
- Add page
- Add text
- Save file



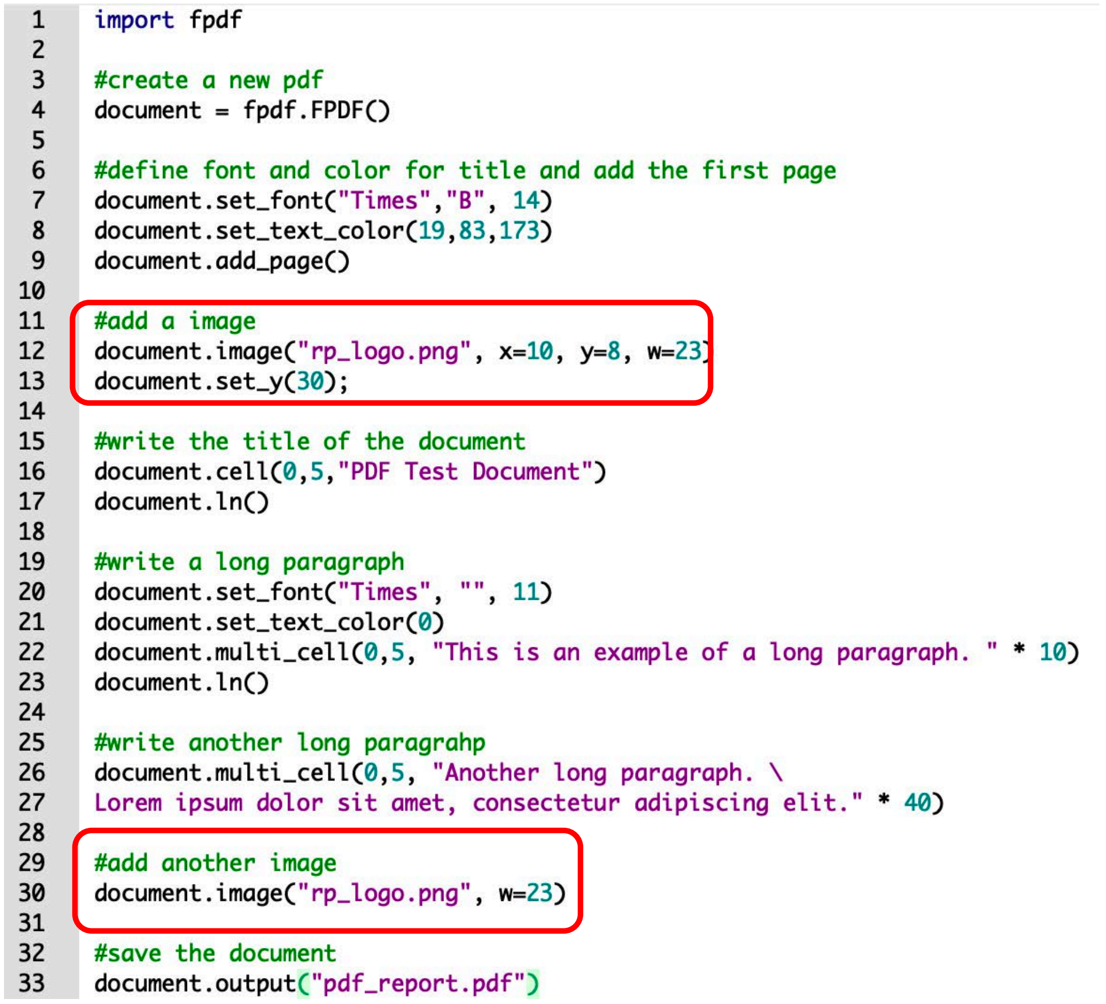
PDF Test Document

This is an example of a long paragraph. This is an example of a long paragraph. This is an example of a long paragraph. This is an example of a long paragraph. This is an example of a long paragraph. This is an example of a long paragraph. This is an example of a long paragraph. This is an example of a long paragraph. This is an example of a long paragraph. This is an example of a long paragraph.

Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.

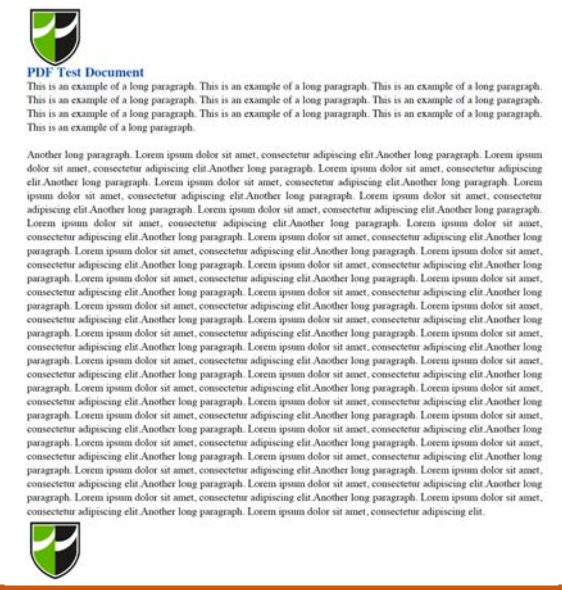
https://pyfpdf.readthedocs.io/en/latest/reference/image/index.html

PDF – adding images



```
import fpdf

#create a new pdf
document = fpdf.FPDF()

#define font and color for title and add the first page
document.set_font("Times","B", 14)
document.set_text_color(19,83,173)
document.add_page()

#add a image
document.image("rp_logo.png", x=10, y=8, w=23)
document.set_y(30);

#write the title of the document
document.cell(0,5,"PDF Test Document")
document.ln()

#write a long paragraph
document.set_font("Times", "", 11)
document.set_text_color(0)
document.multi_cell(0,5, "This is an example of a long paragraph. " * 10)
document.ln()

#write another long paragrahp
document.multi_cell(0,5, "Another long paragraph. \
Lorem ipsum dolor sit amet, consectetur adipiscing elit." * 40)

#add another image
document.image("rp_logo.png", w=23)

#save the document
document.output("pdf_report.pdf")
```

- Import fpdf
- Create a new pdf document
- Add page
- Add text, logo
- Save file

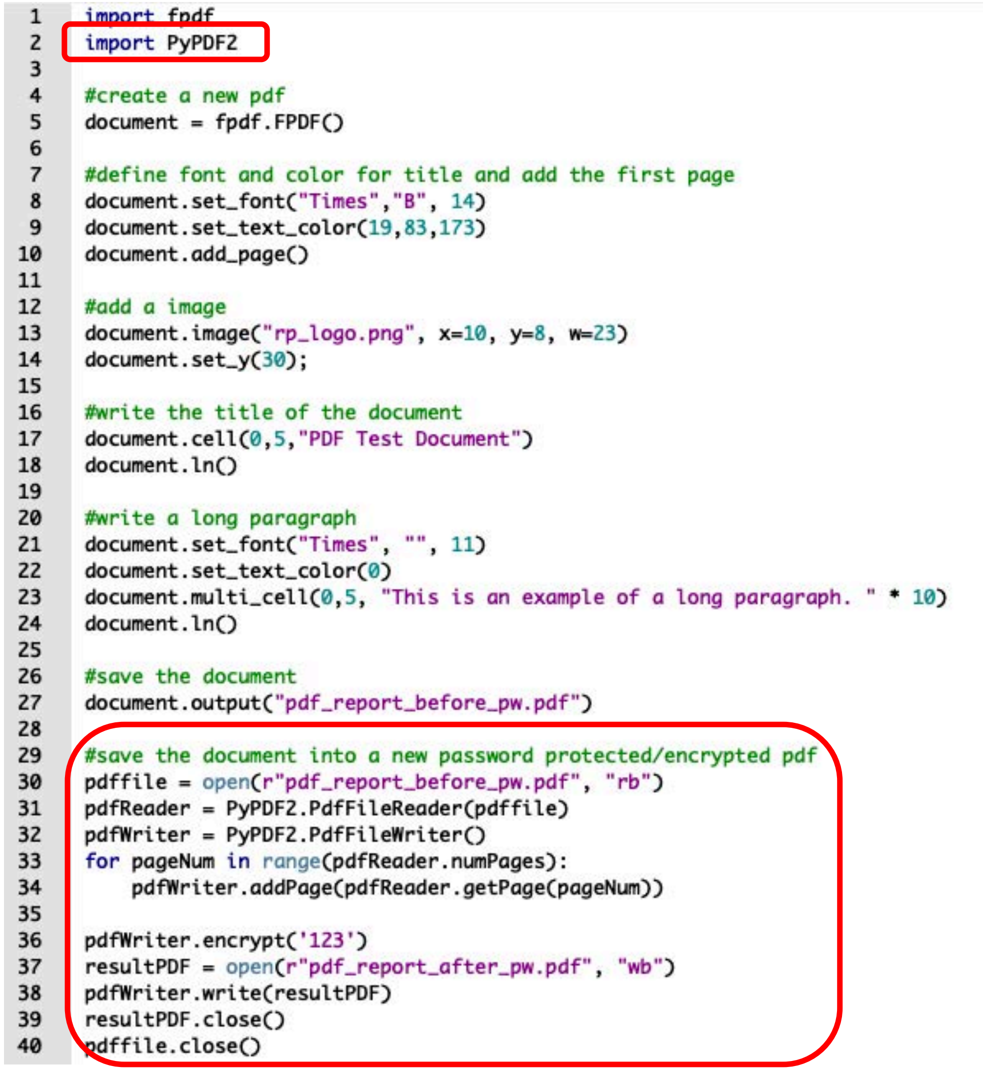


PDF Test Document

This is an example of a long paragraph. This is an example of a long paragraph. This is an example of a long paragraph. This is an example of a long paragraph. This is an example of a long paragraph. This is an example of a long paragraph. This is an example of a long paragraph. This is an example of a long paragraph. This is an example of a long paragraph. This is an example of a long paragraph.

Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.Another long paragraph. Lorem ipsum dolor sit amet, consectetur adipiscing elit.

PDF – Adding password



```
import fpdf
import PyPDF2

#create a new pdf
document = fpdf.FPDF()

#define font and color for title and add the first page
document.set_font("Times","B", 14)
document.set_text_color(19,83,173)
document.add_page()

#add a image
document.image("rp_logo.png", x=10, y=8, w=23)
document.set_y(30);

#write the title of the document
document.cell(0,5,"PDF Test Document")
document.ln()

#write a long paragraph
document.set_font("Times", "", 11)
document.set_text_color(0)
document.multi_cell(0,5, "This is an example of a long paragraph. " * 10)
document.ln()

#save the document
document.output("pdf_report_before_pw.pdf")

#save the document into a new password protected/encrypted pdf
pdffile = open(r"pdf_report_before_pw.pdf", "rb")
pdfReader = PyPDF2.PdfFileReader(pdffile)
pdfWriter = PyPDF2.PdfFileWriter()
for pageNum in range(pdfReader.numPages):
    pdfWriter.addPage(pdfReader.getPage(pageNum))

pdfWriter.encrypt('123')
resultPDF = open(r"pdf_report_after_pw.pdf", "wb")
pdfWriter.write(resultPDF)
resultPDF.close()
pdffile.close()
```

- pip install PyPDF2

https://pythonhosted.org/PyPDF2/